F 1

```
   CCCCCCCCCCCC     OOOOOOOOO     PPPPPPPPPPPP     YYY         YYY
   CCCCCCCCCCCC     OOOOOOOOO     PPPPPPPPPPPP     YYY         YYY
   CCCCCCCCCCCC     OOOOOOOOO     PPPPPPPPPPPP     YYY         YYY
CCC              OOO         OOO  PPP         PPP  YYY         YYY
CCC              OOO         OOO  PPP         PPP  YYY         YYY
CCC              OOO         OOO  PPP         PPP  YYY         YYY
CCC              OOO         OOO  PPP         PPP     YYY   YYY
CCC              OOO         OOO  PPP         PPP     YYY   YYY
CCC              OOO         OOO  PPP         PPP     YYY   YYY
CCC              OOO         OOO  PPPPPPPPPPPP           YYY
CCC              OOO         OOO  PPPPPPPPPPPP           YYY
CCC              OOO         OOO  PPPPPPPPPPPP           YYY
CCC              OOO         OOO  PPP                    YYY
CCC              OOO         OOO  PPP                    YYY
CCC              OOO         OOO  PPP                    YYY
CCC              OOO         OOO  PPP                    YYY
CCC              OOO         OOO  PPP                    YYY
CCC              OOO         OOO  PPP                    YYY
   CCCCCCCCCCCC     OOOOOOOOO     PPP                    YYY
   CCCCCCCCCCCC     OOOOOOOOO     PPP                    YYY
   CCCCCCCCCCCC     OOOOOOOOO     PPP                    YYY
```

```
                                                                         K  3
**FILE**ID**COPYCLI
```

```
  CCCCCCCC    OOOOOO    PPPPPPPP    YY      YY    CCCCCCCC  LL            IIIIII
  CCCCCCCC    OOOOOO    PPPPPPPP    YY      YY    CCCCCCCC  LL            IIIIII
CC          OO      OO  PP      PP  YY      YY  CC          LL              II
CC          OO      OO  PP      PP  YY      YY  CC          LL              II
CC          OO      OO  PP      PP    YY  YY    CC          LL              II
CC          OO      OO  PP      PP    YY  YY    CC          LL              II
CC          OO      OO  PPPPPPPP        YY      CC          LL              II
CC          OO      OO  PPPPPPPP        YY      CC          LL              II
CC          OO      OO  PP              YY      CC          LL              II
CC          OO      OO  PP              YY      CC          LL              II
CC          OO      OO  PP              YY      CC          LL              II          ....
CC          OO      OO  PP              YY      CC          LL              II          ....
  CCCCCCCC    OOOOOO    PP              YY        CCCCCCCC  LLLLLLLLLL    IIIIII          ....
  CCCCCCCC    OOOOOO    PP              YY        CCCCCCCC  LLLLLLLLLL    IIIIII          ....

LL            IIIIII      SSSSSSSS
LL            IIIIII      SSSSSSSS
LL              II      SS
LL              II      SS
LL              II      SS
LL              II      SS
LL              II        SSSSSS
LL              II        SSSSSS
LL              II              SS
LL              II              SS
LL              II              SS
LL              II              SS
LLLLLLLLLL    IIIIII    SSSSSSSS
LLLLLLLLLL    IIIIII    SSSSSSSS
```

```
0 MODULE copycli (    ! Declarations of CLI data structures for the COPY command
0                     LANGUAGE (BLISS32),
0                     IDENT = 'V04-000'
0                     ) =
1 BEGIN
1
1 !
1 !****************************************************************************
1 !*                                                                          *
1 !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                 *
1 !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                  *
1 !*  ALL RIGHTS RESERVED.                                                    *
1 !*                                                                          *
1 !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED   *
1 !*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE   *
1 !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER   *
1 !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY   *
1 !*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY   *
1 !*  TRANSFERRED.                                                            *
1 !*                                                                          *
1 !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE   *
1 !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT   *
1 !*  CORPORATION.                                                            *
1 !*                                                                          *
1 !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS   *
1 !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                 *
1 !*                                                                          *
1 !*                                                                          *
1 !****************************************************************************
1
1 !++
1 ! FACILITY:      COPY
1 !
1 ! ABSTRACT:
1 !
1 !       This module contains all the routines for processing the COPY command
1 !       qualifiers.
1 !
1 ! ENVIRONMENT:
1 !
1 !       VAX/VMS operating system, unprivileged user mode utility,
1 !       operates at non-AST level.
1 !
1 !--
1 !++
1 !
1 ! AUTHOR:        Carol Peters,   CREATION DATE:  28 April 1978 07:36
1 !
1 ! REVISION HISTORY:
1 !
1 !       V3-003  TSK0003         Tamar Krichevsky          9-FEB-1984
1 !               Change addressing mode for LIB$CVT_DTB and LIB$LOOKUP_KEY
1 !               to general.
1 !
1 !       V3-002  TSK0002         Tamar Krichevsky         10-Aug-1983
1 !               Fix default for /PROTECTION qualifier so that if fields which
1 !               have not been specified are left alone.
```

```
1 !
1 !     V3-001  TSK0001         Tamar Krichevsky        18-Jan-1983
1 !             Rework whole module.  Change Command Language Interface over
1 !             to new CLI.  Create two global routines: COPY$GET_GLOBAL_QUAL
1 !             and COPY$GET_LOCAL_QUAL.  These routines simulate parts of the
1 !             CLI so that COPY/QUAL a,b/NOQUAL,c * and COPY a,b/NOQUAL,c */QUAL
1 !             behave the same.
1 !
1 !             Add the common qualifiers (/BEFORE, /SINCE, /CREATED, /MODIFIED
1 !             /BACKUP, /EXPIRED, /EXCLUDE, /BY_OWNER AND /CONFIRM).
1 !
1 !     003     TMH0003         T. Halvorsen    17-Nov-1979
1 !             Add cleanup2_desc for output parameter cleanup call.
1 !
1 !     002     TMH0002         T. Halvorsen    25-Jul-1979
1 !             Add relative volume placement control
1 !--
```

```
  1 !
  1 ! Table of contents
  1 !
  1
  1 FORWARD ROUTINE
  1     copy$get_global_qual: NOVALUE,                          ! Get global command qualifiers
  1     copy$get_local_qual : NOVALUE,                          ! Get local command qualifiers
  1     protection_parse    : NOVALUE,                          ! Parse routine for /PROTECTION qualifier
  1     parse_protection_value : NOVALUE;                       ! Parse the /PROTECTION keyword values (RWED)
  1
  1 !
  1 ! Include files
  1 !
  1
  1 LIBRARY 'SYS$LIBRARY:STARLET.L32';                           ! Common system definitions
  1 REQUIRE 'SRC$:COPYMSG.REQ';                                 ! Put message macros
  1
  1 !
  1 ! Literals
  1 !
  1
  1 BIND
  1
  1     ! Descriptors for the qualifier names, used while parsing the command line.
  1     !
  1     verb_desc            = $DESCRIPTOR('$VERB'),
  1     log_msg_desc         = $DESCRIPTOR('LOG'),
  1     concatenate_desc     = $DESCRIPTOR('CONCATENATE'),
  1     new_version_desc     = $DESCRIPTOR('NEW_VERSION'),
  1     allocation_desc      = $DESCRIPTOR('ALLOCATION'),
  1     contiguous_desc      = $DESCRIPTOR('CONTIGUOUS'),
  1     extension_desc       = $DESCRIPTOR('EXTENSION'),
  1     file_max_desc        = $DESCRIPTOR('FILE_MAXIMUM'),
  1     protection_desc      = $DESCRIPTOR('PROTECTION'),
  1     read_check_desc      = $DESCRIPTOR('READ_CHECK'),
  1     write_check_desc     = $DESCRIPTOR('WRITE_CHECK'),
  1     overlay_desc         = $DESCRIPTOR('OVERLAY'),
  1     volume_desc          = $DESCRIPTOR('VOLUME'),
  1     truncate_desc        = $DESCRIPTOR('TRUNCATE'),
  1     replace_desc         = $DESCRIPTOR('REPLACE')
  1     ;
  1
  1 !
  1 ! Macros
  1 !
  1
  1 MACRO
  1
  1     ! These macros are all used in processing the /PROTECTION qualifier.
  1     !
M 1     BIT_LOCATION( L, B, S, X) =                     ! Extract a bit from a field definition
  1             B %,
M 1     PROT_MASK(DISP,SIZE) =                          ! XAB$W_PRO bit and mask definitions macros
  1             MASK_DEF(XAB$W_PRO,DISP,SIZE) %,
M 1     MASK_DEF(L,B,S,X,DISP,SIZE) =
  1             0, B+DISP, SIZE, X %;
  1
```

```
;  133          0212   1 !
;  134          0213   1 ! External declarations
;  135          0214   1 !
;  136          0215   1
;  137          0216   1 EXTERNAL
;  138          0217   1     copy$prot_value,                                      ! Protection keyword value table
;  139          0218   1     copy$cli_status     : $BBLOCK,                        ! Results of the command line parse
;  140          0219   1     copy$sem_status     : $BBLOCK                         ! Semantics for copy operation
;  141          0220   1     ;
;  142          0221   1
;  143          0222   1 REQUIRE
;  144          0223   1     'SRC$:COPY.REQ';                                      ! Field definitions for COPY$CLI_STATUS and COPY$SEM
; %PRINT:         File: VMSMAC.B32, Version V04-000, Edit 1, WWC, 09-JAN-1978
;  145          0678  1
;  146          0679  1
;  147          0680  1 EXTERNAL ROUTINE
;  148          0681  1     cli$present,                                  ! Determine if a qualifier appears on the command li
;  149          0682  1     cli$get_value,                                ! Retrieve the qualifier's value
;  150          0683  1     lib$qual_file_parse,                          ! Parse the common file qualifiers
;  151          0684  1     lib$cvt_dtb : ADDRESSING_MODE(GENERAL),       ! Convert String to binary
;  152          0685  1     lib$lookup_key: ADDRESSING_MODE(GENERAL);     ! Library keyword lookup routine
;   154          0686  1 !
;   155          0687  1 ! Global variables
;   156          0688  1 !
;   157          0689  1
;   158          0690  1 GLOBAL
;   159          0691  1
;   160          0692  1 !
;   161          0693  1 ! The following variables hold the current qualifier and option values gathered during the
;   162          0694  1 ! CLI processing.  These values may change as local qualifiers are parsed.  The global
;   163          0695  1 ! values are stored in COPY$CLI_STATUS.
;   164          0696  1 !
;   165          0697  1
;   166          0698  1     common_qual_context,                                        ! Common qualifier data area
;   167          0699  1     curr_allocation_value,                                      ! Binary allocation value
;   168          0700  1     curr_extension_value,                                       ! Binary extension value
;   169          0701  1     curr_file_max_value,                                        ! Binary file maximum value
;   170          0702  1     curr_protection_or  :                                       ! Protection mask to set bits
;   171          0703  1         $BBLOCK[ 2 ]
;   172          0704  1         INITIAL (REP 2 OF BYTE (0)),
;   173          0705  1     curr_protection_and :                                       ! Protection mask to clear bits
;   174          0706  1         $BBLOCK[ 2 ]
;   175          0707  1         INITIAL (REP 2 OF BYTE (-1)),
;   176          0708  1     curr_volume_value   : INITIAL (0)                           ! Relative volume number
;   177          0709  1     ;
;   178          0710  1
```

```
1 GLOBAL ROUTINE COPY$GET_GLOBAL_QUAL:  NOVALUE =          ! Retrieve gloabl qualifiers from the CLI
1
1 !++
1 ! FUNCTIONAL DESCRIPTION:
1 !
1 !       This routine retrieves the command level qualifiers from the
1 !       Command Language Interpreter.  It treats any qualifiers found
1 !       as global, even if they are only locally present.  This ensures
1 !       that qualifiers which appear on the output file have the same
1 !       effect as ones which appear on the verb.
1 !
1 !
1 ! FORMAL PARAMETERS:
1 !
1 !       None
1 !
1 ! IMPLICIT INPUTS:
1 !
1 !       None
1 !
1 ! IMPLICIT OUTPUTS:
1 !
1 !       COPY$CLI_STATUS - Relevant command and qualifier indicators set
1 !
1 ! ROUTINE VALUE:
1 !
1 !       None
1 !
1 ! SIDE EFFECTS:
1 !
1 !       None
1 !
1 !--
1
2     BEGIN
2
2     LOCAL
2         common_qual_flags,                               ! Bits which select the common qualifiers to be pars
2         rtn_status,                                      ! Status returned from external calls
2         cli_desc :  $BBLOCK[ dsc$c_s_bln ]               ! Dynamic string descriptor, points to values
2         ;                                                !    returned from calls to the CLI
2
2
2
2     ! Initialize descriptor.
2     !
2     CH$FILL( 0, DSC$C_S_BLN, cli_desc);
2     cli_desc[ DSC$B_CLASS ] = DSC$K_CLASS_D;
2
2
2     ! Retrieve the verb from the command line.  Determine if it is a COPY or APPEND command.
2     !
2     CLI$GET_VALUE( verb_desc, cli_desc);
2     IF CH$RCHAR( .cli_desc[ DSC$A_POINTER ]) EQL 'A'
2     THEN
3         BEGIN
```

```
;  237      0768  3
;  238      0769  3            ! It was an APPEND command.  Set the append command flag and parse the APPEND
;  239      0770  3            ! specific qualifiers.
;  240      0771  3            !
;  241      0772  3            append_command   = TRUE;
;  242      0773  3            new_version_qual = CLI$PRESENT( new_version_desc );
;  243      0774  3            END
;  244      0775  2        ELSE
;  245      0776  3            BEGIN
;  246      0777  3
;  247      0778  3            ! It was a COPY command.  Parse the COPY specific qualifiers.
;  248      0779  3            !
;  249      0780  3            ! /CONCATENATE, /TRUNCATE -- Set the apropriate flags if the qualifier
;  250      0781  3            ! was given or negated.
;  251      0782  3            !
;  252      0783  3            SELECTONE CLI$PRESENT( concatenate_desc ) OF
;  253      0784  3            SET
;  254      0785  4                [ CLI$_PRESENT ] : BEGIN
;  255      0786  4                                   concat_qual          = TRUE;
;  256      0787  4                                   explicit_concat_qual = TRUE;
;  257      0788  3                                   END;
;  258      0789  3                [ CLI$_NEGATED ] : negated_concat_qual  = TRUE;
;  259      0790  3            TES;
;  260      0791  3
;  261      0792  3            SELECTONE CLI$PRESENT( truncate_desc ) OF
;  262      0793  3            SET
;  263      0794  3                [ CLI$_PRESENT,
;  264      0795  3                  CLI$_LOCPRES ] : truncate_qual    = TRUE;
;  265      0796  3                [ CLI$_NEGATED ] : truncate_negated = TRUE;
;  266      0797  3            TES;
;  267      0798  3
;  268      0799  3            ! /OVERLAY and  /REPLACE
;  269      0800  3            !
;  270      0801  3            overlay_qual = CLI$PRESENT( overlay_desc );
;  271      0802  3            replace_qual = CLI$PRESENT( replace_desc );
;  272      0803  3
;  273      0804  3
;  274      0805  3            ! /VOLUME
;  275      0806  3            !
;  276      0807  4            IF (volume_qual = CLI$PRESENT( volume_desc ))
;  277      0808  3            THEN
;  278      0809  4                BEGIN
;  279      0810  4
;  280      0811  4                ! Get the value and and convert it from a string into binary.
;  281      0812  4                !
;  282      0813  4                CLI$GET_VALUE( volume_desc, cli_desc );
;  283      0814  5                IF NOT (rtn_status = LIB$CVT_DTB(.cli_desc[ DSC$W_LENGTH ],
;  284      0815  5                                    .cli_desc[ DSC$A_POINTER ], volume_value))
;  285      0816  4                THEN
;  286      0817  4                    PUT_MESSAGEX( MSG$_INVQUAVAL, 2, cli_desc, volume_desc );
;  287      0818  4                curr_volume_value = .volume_value;
;  288      0819  3                END;
;  289      0820  2            END;
;  290      0821  2
;  291      0822  2        ! Parse the qualifiers which are applicable to both commands.  First,
;  292      0823  2        ! the common qualifiers (/CONFIRM, /BEFORE, /SINCE, /EXCLUDE, /CREATED,
;  293      0824  2        ! /MODIFIED, /BACKUP, /EXPIRED, /BY_OWNER)
```

```
2       !
2       ! Initialize the flags longword so that all of the common qualifiers will
2       ! be parsed.  Then, parse the qualifiers.
2       !
2       common_qual_flags = LIB$M_CQF_CONFIRM  OR  LIB$M_CQF_BEFORE  OR
2                           LIB$M_CQF_SINCE    OR  LIB$M_CQF_CREATED OR
2                           LIB$M_CQF_MODIFIED OR  LIB$M_CQF_BACKUP  OR
2                           LIB$M_CQF_EXPIRED  OR  LIB$M_CQF_EXCLUDE OR
2                           LIB$M_CQF_BYOWNER;
2
3       IF NOT (rtn_status = LIB$QUAL_FILE_PARSE( common_qual_flags, common_qual_context ))
2       THEN
2           PUT_MESSAGEX( .rtn_status );
2
2
2       ! /LOG, /READ_CHECK, /WRITE_CHECK and /CONTIGUOUS
2       !
2       log_msg_qual  = CLI$PRESENT( log_msg_desc );
2
2       read_chk_qual = CLI$PRESENT( read_check_desc );
2
2       SELECTONE CLI$PRESENT( write_check_desc ) OF
2       SET
2           [ CLI$_PRESENT,
2             CLI$_LOCPRES ] : write_chk_qual    = TRUE;
2           [ CLI$_NEGATED ] : write_chk_negated = TRUE;
2       TES;
2       write_chk_qual = CLI$PRESENT( write_check_desc );
2
2       SELECTONE CLI$PRESENT( contiguous_desc ) OF
2       SET
2           [ CLI$_PRESENT,
2             CLI$_LOCPRES ] : contig_qual    = TRUE;
2           [ CLI$_NEGATED ] : contig_negated = TRUE;
2       TES;
2
2
2       ! /ALLOCATION
2       !
3       IF (alloc_qual = CLI$PRESENT( allocation_desc ))
3       THEN
3           BEGIN
3
3           ! Get the value and and convert it from a string into binary.
3           !
3           CLI$GET_VALUE( allocation_desc, cli_desc );
4           IF NOT (rtn_status = LIB$CVT_DTB(.cli_desc[ DSC$W_LENGTH ],
4                               .cli_desc[ DSC$A_POINTER ], allocation_value))
3           THEN
3               PUT_MESSAGEX( MSG$_INVQUAVAL, 2, cli_desc, allocation_desc );
3           curr_allocation_value = .allocation_value;
2           END;
2
2
2       ! /EXTENSION
2       !
3       IF (extend_qual = CLI$PRESENT( extension_desc ))
```

```
;  351          0882  2       THEN
;  352          0883  3           BEGIN
;  353          0884  3
;  354          0885  3           ! Get the value and and convert it from a string into binary.
;  355          0886  3           !
;  356          0887  3           CLI$GET_VALUE( extension_desc, cli_desc );
;  357          0888  4           IF NOT (rtn_status = LIB$CVT_DTB(.cli_desc[ DSC$W_LENGTH ],
;  358          0889  4                               .cli_desc[ DSC$A_POINTER ], extension_value))
;  359          0890  3           THEN
;  360          0891  3               PUT_MESSAGEX( MSG$_INVQUAVAL, 2, cli_desc, extension_desc );
;  361          0892  3           curr_extension_value = .extension_value;
;  362          0893  2           END;
;  363          0894  2
;  364          0895  2
;  365          0896  2       ! /FILE_MAXIMUM
;  366          0897  2       !
;  367          0898  3       IF (file_max_qual = CLI$PRESENT( file_max_desc ))
;  368          0899  2       THEN
;  369          0900  3           BEGIN
;  370          0901  3
;  371          0902  3           ! Get the value and and convert it from a string into binary.
;  372          0903  3           !
;  373          0904  3           CLI$GET_VALUE( file_max_desc, cli_desc );
;  374          0905  4           IF NOT (rtn_status = LIB$CVT_DTB(.cli_desc[ DSC$W_LENGTH ],
;  375          0906  4                               .cli_desc[ DSC$A_POINTER ], file_max_value))
;  376          0907  3           THEN
;  377          0908  3               PUT_MESSAGEX( MSG$_INVQUAVAL, 2, cli_desc, file_max_desc );
;  378          0909  3           curr_file_max_value = .file_max_value;
;  379          0910  2           END;
;  380          0911  2
;  381          0912  2
;  382          0913  2       ! /PROTECTION
;  383          0914  2       !
;  384          0915  3       IF (protect_qual = CLI$PRESENT( protection_desc ))
;  385          0916  2       THEN
;  386          0917  3           BEGIN
;  387          0918  3
;  388          0919  3           ! Parse the keyword value and save the results.
;  389          0920  3           !
;  390          0921  3           protection_parse();
;  391          0922  3           protection_and = .(curr_protection_and);
;  392          0923  3           protection_or  = .(curr_protection_or);
;  393          0924  2           END;
;  394          0925  1       END;                                        ! routine COPY$GET_GLOBAL_QUAL

                                                                    .TITLE  COPYCLI
                                                                    .IDENT  \V04-000\

                                                                    .PSECT  $PLIT$,NOWRT,NOEXE,2

                          42  52  45  56  24  00000 P.AAB:          .ASCII  \$VERB\                                  ;
                                              00005                 .BLKB   3
                                    00000005  00008 P.AAA:          .LONG   5                                        ;
                                   00000000'  0000C                 .ADDRESS P.AAB                                   ;
                                  47  4F  4C  00010 P.AAD:          .ASCII  \LOG\                                    ;
                                              00013                 .BLKB   1
```

```
                                         00000003  00014 P.AAC:    .LONG    3                          ;
                                         00000000' 00018           .ADDRESS P.AAD                      ;
          45 54 41 4E 45 54 41 43 4E 4F 43         0001C P.AAF:    .ASCII   \CONCATENATE\              ;
                                                   00027           .BLKB    1
                                         0000000B  00028 P.AAE:    .LONG    11                         ;
                                         00000000' 0002C           .ADDRESS P.AAF                      ;
          4E 4F 49 53 52 45 56 5F 57 45 4E         00030 P.AAH:    .ASCII   \NEW_VERSION\              ;
                                                   0003B           .BLKB    1
                                         0000000B  0003C P.AAG:    .LONG    11                         ;
                                         00000000' 00040           .ADDRESS P.AAH                      ;
             4E 4F 49 54 41 43 4F 4C 4C 41         00044 P.AAJ:    .ASCII   \ALLOCATION\               ;
                                                   0004E           .BLKB    2
                                         0000000A  00050 P.AAI:    .LONG    10                         ;
                                         00000000' 00054           .ADDRESS P.AAJ                      ;
             53 55 4F 55 47 49 54 4E 4F 43         00058 P.AAL:    .ASCII   \CONTIGUOUS\               ;
                                                   00062           .BLKB    2
                                         0000000A  00064 P.AAK:    .LONG    10                         ;
                                         00000000' 00068           .ADDRESS P.AAL                      ;
                4E 4F 49 53 4E 45 54 58 45         0006C P.AAN:    .ASCII   \EXTENSION\                ;
                                                   00075           .BLKB    3
                                         00000009  00078 P.AAM:    .LONG    9                          ;
                                         00000000' 0007C           .ADDRESS P.AAN                      ;
       4D 55 4D 49 58 41 4D 5F 45 4C 49 46         00080 P.AAP:    .ASCII   \FILE_MAXIMUM\             ;
                                         0000000C  0008C P.AAO:    .LONG    12                         ;
                                         00000000' 00090           .ADDRESS P.AAP                      ;
             4E 4F 49 54 43 45 54 4F 52 50         00094 P.AAR:    .ASCII   \PROTECTION\               ;
                                                   0009E           .BLKB    2
                                         0000000A  000A0 P.AAQ:    .LONG    10                         ;
                                         00000000' 000A4           .ADDRESS P.AAR                      ;
             4B 43 45 48 43 5F 44 41 45 52         000A8 P.AAT:    .ASCII   \READ_CHECK\               ;
                                                   000B2           .BLKB    2
                                         0000000A  000B4 P.AAS:    .LONG    10                         ;
                                         00000000' 000B8           .ADDRESS P.AAT                      ;
          4B 43 45 48 43 5F 45 54 49 52 57         000BC P.AAV:    .ASCII   \WRITE_CHECK\              ;
                                                   000C7           .BLKB    1
                                         0000000B  000C8 P.AAU:    .LONG    11                         ;
                                         00000000' 000CC           .ADDRESS P.AAV                      ;
                         59 41 4C 52 45 56 4F      000D0 P.AAX:    .ASCII   \OVERLAY\                  ;
                                                   000D7           .BLKB    1
                                         00000007  000D8 P.AAW:    .LONG    7                          ;
                                         00000000' 000DC           .ADDRESS P.AAX                      ;
                            45 4D 55 4C 4F 56      000E0 P.AAZ:    .ASCII   \VOLUME\                   ;
                                                   000E6           .BLKB    2
                                         00000006  000E8 P.AAY:    .LONG    6                          ;
                                         00000000' 000EC           .ADDRESS P.AAZ                      ;
                      45 54 41 43 4E 55 52 54      000F0 P.ABB:    .ASCII   \TRUNCATE\                 ;
                                         00000008  000F8 P.ABA:    .LONG    8                          ;
                                         00000000' 000FC           .ADDRESS P.ABB                      ;
                         45 43 41 4C 50 45 52      00100 P.ABD:    .ASCII   \REPLACE\                  ;
                                                   00107           .BLKB    1
                                         00000007  00108 P.ABC:    .LONG    7                          ;
                                         00000000' 0010C           .ADDRESS P.ABD                      ;

                                                                   .PSECT   $GLOBAL$,NOEXE,2

                                                   00000 COMMON_QUAL_CONTEXT::
                                                                   .BLKB    4
```

```
                                                            00004 CURR_ALLOCATION_VALUE::
                                                                          .BLKB   4
                                                            00008 CURR_EXTENSION_VALUE::
                                                                          .BLKB   4
                                                            0000C CURR_FILE_MAX_VALUE::
                                                                          .BLKB   4
                                                        00# 00010 CURR_PROTECTION_OR::
                                                                          .BYTE   0[2]                                             ;
                                                            00012         .BLKB   2
                                                        FF# 00014 CURR_PROTECTION_AND::
                                                                          .BYTE   -1[2]                                            ;
                                                            00016         .BLKB   2
                                                  00000000  00018 CURR_VOLUME_VALUE::
                                                                          .LONG   0                                                ;

                                                                  VERB_DESC=          P.AAA
                                                                  LOG_MSG_DESC=       P.AAC
                                                                  CONCATENATE_DESC=   P.AAE
                                                                  NEW_VERSION_DESC=   P.AAG
                                                                  ALLOCATION_DESC=    P.AAI
                                                                  CONTIGUOUS_DESC=    P.AAK
                                                                  EXTENSION_DESC=     P.AAM
                                                                  FILE_MAX_DESC=      P.AAO
                                                                  PROTECTION_DESC=    P.AAQ
                                                                  READ_CHECK_DESC=    P.AAS
                                                                  WRITE_CHECK_DESC=   P.AAU
                                                                  OVERLAY_DESC=       P.AAW
                                                                  VOLUME_DESC=        P.AAY
                                                                  TRUNCATE_DESC=      P.ABA
                                                                  REPLACE_DESC=       P.ABC
                                                                          .EXTRN  COPY$MSG_NUMBER
                                                                          .EXTRN  COPY$PROT_VALUE
                                                                          .EXTRN  COPY$CLI_STATUS
                                                                          .EXTRN  COPY$SEM_STATUS
                                                                          .EXTRN  CLI$_PRESENT, CLI$_NEGATED
                                                                          .EXTRN  CLI$_LOCPRES, CLI$_LOCNEG
                                                                          .EXTRN  CLI$PRESENT, CLI$GET_VALUE
                                                                          .EXTRN  LIB$QUAL_FILE_PARSE
                                                                          .EXTRN  LIB$CVT_DTB, LIB$LOOKUP_KEY

                                                                          .PSECT  $CODE$,NOWRT,2

                                                       0FFC 00000         .ENTRY  COPY$GET_GLOBAL_QUAL, Save R2,R3,R4,R5,R6,- ; 0711
                                                                                  R7,R8,R9,R10,R11                              ;
                                       5B 00000000G  00  9E 00002         MOVAB   LIB$STOP, R11                                 ;
                                       5A 00000000G  00  9E 00009         MOVAB   LIB$SIGNAL, R10                               ;
                                       59      0000G  CF  9E 00010         MOVAB   COPY$MSG_NUMBER, R9                           ;
                                       58      0000G  CF  9E 00015         MOVAB   CLI$PRESENT, R8                               ;
                                       57      0000'  CF  9E 0001A         MOVAB   VOLUME_DESC, R7                               ;
                                       56      0000G  CF  9E 0001F         MOVAB   COPY$CLI_STATUS+4, R6                         ;
                                       5E             0C  C2 00024         SUBL2   #12, SP                                       ;
          08             00            6E             00  2C 00027         MOVC5   #0, (SP), #0, #8, CLI_DESC                    ; 0758
                                                 04  AE     0002C                                                               ;
                                  07   AE             02  90 0002E         MOVB    #2, CLI_DESC+3                                ; 0759
                                                 04  AE  9F 00032         PUSHAB  CLI_DESC                                      ; 0764
                                               FF20  C7  9F 00035         PUSHAB  VERB_DESC                                     ;
                                0000G  CF             02  FB 00039         CALLS   #2, CLI$GET_VALUE                             ;
```

```
                    41  8F      08  BE  91 0003E            CMPB    @CLI_DESC+4, #65                         ; 0765
                                    14  12 00043            BNEQ    1$                                       ;
                    FC  A6          01  88 00045            BISB2   #1, COPY$CLI_STATUS                      ; 0772
                              FF54  C7  9F 00049            PUSHAB  NEW_VERSION_DESC                         ; 0773
                        68          01  FB 0004D            CALLS   #1, CLI$PRESENT                          ;
FC    A6            01  04          50  F0 00050            INSV    R0, #4, #1, COPY$CLI_STATUS              ;
                                  00E1  31 00056            BRW     9$                                       ; 0765
                              FF40  C7  9F 00059 1$:        PUSHAB  CONCATENATE_DESC                         ; 0783
                        68          01  FB 0005D            CALLS   #1, CLI$PRESENT                          ;
             00000000G  8F          50  D1 00060            CMPL    R0, #CLI$_PRESENT                        ; 0785
                                    0B  12 00067            BNEQ    2$                                       ;
                 0000G  CF          01  88 00069            BISB2   #1, COPY$SEM_STATUS                      ; 0786
                    FC  A6          04  88 0006E            BISB2   #4, COPY$CLI_STATUS                      ; 0787
                                    0D  11 00072            BRB     3$                                       ; 0783
             00000000G  8F          50  D1 00074 2$:        CMPL    R0, #CLI$_NEGATED                        ; 0789
                                    04  12 0007B            BNEQ    3$                                       ;
                    FC  A6          08  88 0007D            BISB2   #8, COPY$CLI_STATUS                      ;
                                10  A7  9F 00081 3$:        PUSHAB  TRUNCATE_DESC                            ; 0792
                        68          01  FB 00084            CALLS   #1, CLI$PRESENT                          ;
             00000000G  8F          50  D1 00087            CMPL    R0, #CLI$_PRESENT                        ; 0794
                                    09  13 0008E            BEQL    4$                                       ;
             00000000G  8F          50  D1 00090            CMPL    R0, #CLI$_LOCPRES                        ;
                                    06  12 00097            BNEQ    5$                                       ;
                    01  A6          20  88 00099 4$:        BISB2   #32, COPY$CLI_STATUS+5                   ; 0795
                                    0E  11 0009D            BRB     6$                                       ;
             00000000G  8F          50  D1 0009F 5$:        CMPL    R0, #CLI$_NEGATED                        ; 0796
                                    05  12 000A6            BNEQ    6$                                       ;
                    01  A6      40  8F  88 000A8            BISB2   #64, COPY$CLI_STATUS+5                   ;
                                F0  A7  9F 000AD 6$:        PUSHAB  OVERLAY_DESC                             ; 0801
                        68          01  FB 000B0            CALLS   #1, CLI$PRESENT                          ;
      66            01  07          50  F0 000B3            INSV    R0, #7, #1, COPY$CLI_STATUS+4            ;
                                20  A7  9F 000B8            PUSHAB  REPLACE_DESC                             ; 0802
                        68          01  FB 000BB            CALLS   #1, CLI$PRESENT                          ;
02    A6            01  01          50  F0 000BE            INSV    R0, #1, #1, COPY$CLI_STATUS+6            ;
                                    57  DD 000C4            PUSHL   R7                                       ; 0807
                        68          01  FB 000C6            CALLS   #1, CLI$PRESENT                          ;
01    A6            01  02          50  F0 000C9            INSV    R0, #2, #1, COPY$CLI_STATUS+5            ;
                        68          50  E9 000CF            BLBC    R0, 9$                                   ;
                                04  AE  9F 000D2            PUSHAB  CLI_DESC                                 ; 0813
                                    57  DD 000D5            PUSHL   R7                                       ;
                 0000G  CF          02  FB 000D7            CALLS   #2, CLI$GET_VALUE                        ;
                                14  A6  9F 000DC            PUSHAB  COPY$CLI_STATUS+24                       ; 0815
                                0C  AE  DD 000DF            PUSHL   CLI_DESC+4                               ;
                        7E      0C  AE  3C 000E2            MOVZWL  CLI_DESC, -(SP)                          ; 0814
             00000000G  00          03  FB 000E6            CALLS   #3, LIB$CVT_DTB                          ;
                        52          50  D0 000ED            MOVL    R0, RTN_STATUS                           ;
                        41          52  E8 000F0            BLBS    RTN_STATUS, 8$                           ;
                        7E    132C  8F  3C 000F3            MOVZWL  #4908, -(SP)                             ; 0817
                        69          01  FB 000F8            CALLS   #1, COPY$MSG_NUMBER                      ;
      7E            00  50          01  7A 000FB            EMUL    #1, R0, #0, -(SP)                        ;
      50            50  8E          08  7B 00100            EDIV    #8, (SP)+, R0, R0                        ;
                        04          50  D1 00105            CMPL    R0, #4                                   ;
                                    16  13 00108            BEQL    7$                                       ;
                                    57  DD 0010A            PUSHL   R7                                       ;
                                08  AE  9F 0010C            PUSHAB  CLI_DESC                                 ;
                                    02  DD 0010F            PUSHL   #2                                       ;
                        7E    132C  8F  3C 00111            MOVZWL  #4908, -(SP)                             ;
```

```
                             69          01  FB 00116        CALLS    #1, COPY$MSG_NUMBER                          ;
                                         50  DD 00119        PUSHL    R0                                           ;
                             6A          04  FB 0011B        CALLS    #4, LIB$SIGNAL                               ;
                                         14  11 0011E        BRB      8$                                           ;
                                         57  DD 00120 7$:    PUSHL    R7                                           ;
                                     08  AE  9F 00122        PUSHAB   CLI_DESC                                     ;
                                         02  DD 00125        PUSHL    #2                                           ;
                             7E    132C  8F  3C 00127        MOVZWL   #4908, -(SP)                                 ;
                             69          01  FB 0012C        CALLS    #1, COPY$MSG_NUMBER                          ;
                                         50  DD 0012F        PUSHL    R0                                           ;
                             6B          04  FB 00131        CALLS    #4, LIB$STOP                                 ;
                       0000' CF      14  A6  D0 00134 8$:    MOVL     COPY$CLI_STATUS+24, CURR_VOLUME_VALUE        ; 0818
                             6E    01FF  8F  3C 0013A 9$:    MOVZWL   #511, COMMON_QUAL_FLAGS                      ; 0832
                                  0000'  CF  9F 0013F        PUSHAB   COMMON_QUAL_CONTEXT                          ; 0835
                                     04  AE  9F 00143        PUSHAB   COMMON_QUAL_FLAGS                            ;
                       0000G CF          02  FB 00146        CALLS    #2, LIB$QUAL_FILE_PARSE                      ;
                             52          50  D0 0014B        MOVL     R0, RTN_STATUS                               ;
                             2A          52  E8 0014E        BLBS     RTN_STATUS, 11$                              ;
                                         52  DD 00151        PUSHL    RTN_STATUS                                   ; 0837
                             69          01  FB 00153        CALLS    #1, COPY$MSG_NUMBER                          ;
    7E        00             50          01  7A 00156        EMUL     #1, R0, #0, -(SP)                            ;
    50        50             8E          08  7B 0015B        EDIV     #8, (SP)+, R0, R0                            ;
                             04          50  D1 00160        CMPL     R0, #4                                       ;
                                         0C  13 00163        BEQL     10$                                          ;
                                         52  DD 00165        PUSHL    RTN_STATUS                                   ;
                             69          01  FB 00167        CALLS    #1, COPY$MSG_NUMBER                          ;
                                         50  DD 0016A        PUSHL    R0                                           ;
                             6A          01  FB 0016C        CALLS    #1, LIB$SIGNAL                               ;
                                         0A  11 0016F        BRB      11$                                          ;
                                         52  DD 00171 10$:   PUSHL    RTN_STATUS                                   ;
                             69          01  FB 00173        CALLS    #1, COPY$MSG_NUMBER                          ;
                                         50  DD 00176        PUSHL    R0                                           ;
                             6B          01  FB 00178        CALLS    #1, LIB$STOP                                 ;
                                   FF2C  C7  9F 0017B 11$:   PUSHAB   LOG_MSG_DESC                                 ; 0842
                             68          01  FB 0017F        CALLS    #1, CLI$PRESENT                              ;
FC  A6        01             01          50  F0 00182        INSV     R0, #1, #1, COPY$CLI_STATUS                  ;
                                     CC  A7  9F 00188        PUSHAB   READ_CHECK_DESC                              ; 0844
                             68          01  FB 0018B        CALLS    #1, CLI$PRESENT                              ;
    66        01             00          50  F0 0018E        INSV     R0, #0, #1, COPY$CLI_STATUS+4                ;
                                     E0  A7  9F 00193        PUSHAB   WRITE_CHECK_DESC                             ; 0846
                             68          01  FB 00196        CALLS    #1, CLI$PRESENT                              ;
                   00000000G 8F          50  D1 00199        CMPL     R0, #CLI$_PRESENT                            ; 0848
                                         09  13 001A0        BEQL     12$                                          ;
                   00000000G 8F          50  D1 001A2        CMPL     R0, #CLI$_LOCPRES                            ;
                                         05  12 001A9        BNEQ     13$                                          ;
                             66          08  88 001AB 12$:   BISB2    #8, COPY$CLI_STATUS+4                        ; 0849
                                         0C  11 001AE        BRB      14$                                          ;
                   00000000G 8F          50  D1 001B0 13$:   CMPL     R0, #CLI$_NEGATED                            ; 0850
                                         03  12 001B7        BNEQ     14$                                          ;
                             66          10  88 001B9        BISB2    #16, COPY$CLI_STATUS+4                       ;
                                     E0  A7  9F 001BC 14$:   PUSHAB   WRITE_CHECK_DESC                             ; 0852
                             68          01  FB 001BF        CALLS    #1, CLI$PRESENT                              ;
    66        01             03          50  F0 001C2        INSV     R0, #3, #1, COPY$CLI_STATUS+4                ;
                                   FF7C  C7  9F 001C7        PUSHAB   CONTIGUOUS_DESC                              ; 0854
                             68          01  FB 001CB        CALLS    #1, CLI$PRESENT                              ;
                   00000000G 8F          50  D1 001CE        CMPL     R0, #CLI$_PRESENT                            ; 0856
                                         09  13 001D5        BEQL     15$                                          ;
                       00000000G  8F          50  D1 001D7        CMPL    R0, #CLI$_LOCPRES                            :
                                              06  12 001DE        BNEQ    16$                                          :
                              FE  A6          08  88 001E0 15$:   BISB2   #8, COPY$CLI_STATUS+2                        : 0857
                                              0D  11 001E4        BRB     17$                                          :
                       00000000G  8F          50  D1 001E6 16$:   CMPL    R0, #CLI$_NEGATED                            : 0858
                                              04  12 001ED        BNEQ    17$                                          :
                              FE  A6          10  88 001EF        BISB2   #16, COPY$CLI_STATUS+2                       :
                                        FF68  C7  9F 001F3 17$:   PUSHAB  ALLOCATION_DESC                              : 0864
                                  68          01  FB 001F7        CALLS   #1, CLI$PRESENT                              :
        FE   A6            01     00          50  F0 001FA        INSV    R0, #0, #1, COPY$CLI_STATUS+2                :
                                  6E          50  E9 00200        BLBC    R0, 20$                                      :
                                          04  AE  9F 00203        PUSHAB  CLI_DESC                                     : 0870
                                        FF68  C7  9F 00206        PUSHAB  ALLOCATION_DESC                              :
                           0000G  CF          02  FB 0020A        CALLS   #2, CLI$GET_VALUE                            :
                                          04  A6  9F 0020F        PUSHAB  COPY$CLI_STATUS+8                            : 0872
                                          0C  AE  DD 00212        PUSHL   CLI_DESC+4                                   :
                                  7E      0C  AE  3C 00215        MOVZWL  CLI_DESC, -(SP)                              : 0871
                       00000000G  00          03  FB 00219        CALLS   #3, LIB$CVT_DTB                              :
                                  52          50  D0 00220        MOVL    R0, RTN_STATUS                               :
                                  45          52  E8 00223        BLBS    RTN_STATUS, 19$                              :
                                  7E    132C  8F  3C 00226        MOVZWL  #4908, -(SP)                                 : 0874
                                  69          01  FB 0022B        CALLS   #1, COPY$MSG_NUMBER                          :
             7E            00     50          01  7A 0022E        EMUL    #1, R0, #0, -(SP)                            :
             50            50     8E          08  7B 00233        EDIV    #8, (SP)+, R0, R0                            :
                                  04          50  D1 00238        CMPL    R0, #4                                       :
                                              18  13 0023B        BEQL    18$                                          :
                                        FF68  C7  9F 0023D        PUSHAB  ALLOCATION_DESC                              :
                                          08  AE  9F 00241        PUSHAB  CLI_DESC                                     :
                                              02  DD 00244        PUSHL   #2                                           :
                                  7E    132C  8F  3C 00246        MOVZWL  #4908, -(SP)                                 :
                                  69          01  FB 0024B        CALLS   #1, COPY$MSG_NUMBER                          :
                                              50  DD 0024E        PUSHL   R0                                           :
                                  6A          04  FB 00250        CALLS   #4, LIB$SIGNAL                               :
                                              16  11 00253        BRB     19$                                          :
                                        FF68  C7  9F 00255 18$:   PUSHAB  ALLOCATION_DESC                              :
                                          08  AE  9F 00259        PUSHAB  CLI_DESC                                     :
                                              02  DD 0025C        PUSHL   #2                                           :
                                  7E    132C  8F  3C 0025E        MOVZWL  #4908, -(SP)                                 :
                                  69          01  FB 00263        CALLS   #1, COPY$MSG_NUMBER                          :
                                              50  DD 00266        PUSHL   R0                                           :
                                  6B          04  FB 00268        CALLS   #4, LIB$STOP                                 :
                           0000'  CF      04  A6  D0 0026B 19$:   MOVL    COPY$CLI_STATUS+8, CURR_ALLOCATION_VALUE     : 0875
                                          90  A7  9F 00271 20$:   PUSHAB  EXTENSION_DESC                               : 0881
                                  68          01  FB 00274        CALLS   #1, CLI$PRESENT                              :
        FE   A6            01     07          50  F0 00277        INSV    R0, #7, #1, COPY$CLI_STATUS+2                :
                                  6B          50  E9 0027D        BLBC    R0, 23$                                      :
                                          04  AE  9F 00280        PUSHAB  CLI_DESC                                     : 0887
                                          90  A7  9F 00283        PUSHAB  EXTENSION_DESC                               :
                           0000G  CF          02  FB 00286        CALLS   #2, CLI$GET_VALUE                            :
                                          08  A6  9F 0028B        PUSHAB  COPY$CLI_STATUS+12                           : 0889
                                          0C  AE  DD 0028E        PUSHL   CLI_DESC+4                                   :
                                  7E      0C  AE  3C 00291        MOVZWL  CLI_DESC, -(SP)                              : 0888
                       00000000G  00          03  FB 00295        CALLS   #3, LIB$CVT_DTB                              :
                                  52          50  D0 0029C        MOVL    R0, RTN_STATUS                               :
                                  43          52  E8 0029F        BLBS    RTN_STATUS, 22$                              :
                                  7E    132C  8F  3C 002A2        MOVZWL  #4908, -(SP)                                 : 0891
                                  69          01  FB 002A7        CALLS   #1, COPY$MSG_NUMBER                          :
```

```
            7E          00              50          01  7A  002AA        EMUL    #1, R0, #0, -(SP)
            50          50              8E          08  7B  002AF        EDIV    #8, (SP)+, R0, R0
                                        04          50  D1  002B4        CMPL    R0, #4
                                                    17  13  002B7        BEQL    21$
                                                90  A7  9F  002B9        PUSHAB  EXTENSION_DESC
                                                08  AE  9F  002BC        PUSHAB  CLI_DESC
                                                    02  DD  002BF        PUSHL   #2
                                        7E    132C  8F  3C  002C1        MOVZWL  #4908, -(SP)
                                        69          01  FB  002C6        CALLS   #1, COPY$MSG_NUMBER
                                                    50  DD  002C9        PUSHL   R0
                                        6A          04  FB  002CB        CALLS   #4, LIB$SIGNAL
                                                    15  11  002CE        BRB     22$
                                                90  A7  9F  002D0 21$:   PUSHAB  EXTENSION_DESC
                                                08  AE  9F  002D3        PUSHAB  CLI_DESC
                                                    02  DD  002D6        PUSHL   #2
                                        7E    132C  8F  3C  002D8        MOVZWL  #4908, -(SP)
                                        69          01  FB  002DD        CALLS   #1, COPY$MSG_NUMBER
                                                    50  DD  002E0        PUSHL   R0
                                        6B          04  FB  002E2        CALLS   #4, LIB$STOP
                                 0000'  CF      08  A6  D0  002E5 22$:   MOVL    COPY$CLI_STATUS+12, CURR_EXTENSION_VALUE    ; 0892
                                                A4  A7  9F  002EB 23$:   PUSHAB  FILE_MAX_DESC                               ; 0898
                                        68          01  FB  002EE        CALLS   #1, CLI$PRESENT
        FF  A6          01              02          50  F0  002F1        INSV    R0, #2, #1, COPY$CLI_STATUS+3
                                        6B          50  E9  002F7        BLBC    R0, 26$
                                                04  AE  9F  002FA        PUSHAB  CLI_DESC                                    ; 0904
                                                A4  A7  9F  002FD        PUSHAB  FILE_MAX_DESC
                                 0000G  CF          02  FB  00300        CALLS   #2, CLI$GET_VALUE
                                                0C  A6  9F  00305        PUSHAB  COPY$CLI_STATUS+16                          ; 0906
                                                0C  AE  DD  00308        PUSHL   CLI_DESC+4
                                        7E      0C  AE  3C  0030B        MOVZWL  CLI_DESC, -(SP)                             ; 0905
                             00000000G  00          03  FB  0030F        CALLS   #3, LIB$CVT_DTB
                                        52          50  D0  00316        MOVL    R0, RTN_STATUS
                                        43          52  E8  00319        BLBS    RTN_STATUS, 25$
                                        7E    132C  8F  3C  0031C        MOVZWL  #4908, -(SP)                                ; 0908
                                        69          01  FB  00321        CALLS   #1, COPY$MSG_NUMBER
            7E          00              50          01  7A  00324        EMUL    #1, R0, #0, -(SP)
            50          50              8E          08  7B  00329        EDIV    #8, (SP)+, R0, R0
                                        04          50  D1  0032E        CMPL    R0, #4
                                                    17  13  00331        BEQL    24$
                                                A4  A7  9F  00333        PUSHAB  FILE_MAX_DESC
                                                08  AE  9F  00336        PUSHAB  CLI_DESC
                                                    02  DD  00339        PUSHL   #2
                                        7E    132C  8F  3C  0033B        MOVZWL  #4908, -(SP)
                                        69          01  FB  00340        CALLS   #1, COPY$MSG_NUMBER
                                                    50  DD  00343        PUSHL   R0
                                        6A          04  FB  00345        CALLS   #4, LIB$SIGNAL
                                                    15  11  00348        BRB     25$
                                                A4  A7  9F  0034A 24$:   PUSHAB  FILE_MAX_DESC
                                                08  AE  9F  0034D        PUSHAB  CLI_DESC
                                                    02  DD  00350        PUSHL   #2
                                        7E    132C  8F  3C  00352        MOVZWL  #4908, -(SP)
                                        69          01  FB  00357        CALLS   #1, COPY$MSG_NUMBER
                                                    50  DD  0035A        PUSHL   R0
                                        6B          04  FB  0035C        CALLS   #4, LIB$STOP
                                 0000'  CF      0C  A6  D0  0035F 25$:   MOVL    COPY$CLI_STATUS+16, CURR_FILE_MAX_VALUE     ; 0909
                                                B8  A7  9F  00365 26$:   PUSHAB  PROTECTION_DESC                             ; 0915
                                        68          01  FB  00368        CALLS   #1, CLI$PRESENT
```

```
     FF   A6          01                05             50  F0 0036B        INSV     R0, #5, #1, COPY$CLI_STATUS+3                  ;
                                        11             50  E9 00371        BLBC     R0, 27$                                        ;
                                 0000V  CF             00  FB 00374        CALLS    #0, PROTECTION_PARSE                           ; 0921
                                    12  A6     0000'   CF  B0 00379        MOVW     CURR_PROTECTION_AND, COPY$CLI_STATUS+22        ; 0922
                                    10  A6     0000'   CF  B0 0037F        MOVW     CURR_PROTECTION_OR, COPY$CLI_STATUS+20         ; 0923
                                                           04 00385 27$:   RET                                                     ; 0925

; Routine Size:  902 bytes,    Routine Base:  $CODE$ + 0000
```

```
1 GLOBAL ROUTINE COPY$GET_LOCAL_QUAL:  NOVALUE =          ! Retrieve local qualifiers from the CLI
1
1 !++
1 ! FUNCTIONAL DESCRIPTION:
1 !
1 !       This routine retrieves the local command qualifiers from the command
1 !       line.
1 !
1 ! FORMAL PARAMETERS:
1 !
1 !       None
1 !
1 ! IMPLICIT INPUTS:
1 !
1 !       None
1 !
1 ! IMPLICIT OUTPUTS:
1 !
1 !       COPY$CLI_STATUS - Relevant command and qualifier indicators set
1 !
1 ! ROUTINE VALUE:
1 !
1 !       None
1 !
1 ! SIDE EFFECTS:
1 !
1 !       None
1 !
1 !--
1
2     BEGIN
2
2     LOCAL
2         rtn_status,                                   ! Status returned from external calls
2         cli_desc :  $BBLOCK[ dsc$c_s_bln ]            ! Dynamic string descriptor, points to values
2         ;                                             !   returned from calls to the CLI
2
2     BIND
2
2         MSG_DESC        = $DESCRIPTOR('can''t change quals in the middle of the command')
2         ;
2
2
2
2
2     ! Initialize descriptor.  Also, if a new output file is being created, then
2     ! reset the current qualifier values to the global values.  This insures
2     ! that if a previous local qualifier changed the value and, on this
2     ! iteration, there is no local qualifier, the value used when creating the
2     ! output file will be the one given by the global qualifier, not the
2     ! previous local qualifier.
2     !
2     CH$FILL( 0, DSC$C_S_BLN, cli_desc);
2     cli_desc[ DSC$B_CLASS ] = DSC$K_CLASS_D;
2
2     IF not .outfile_open
2     THEN
```

```
3          BEGIN
3          curr_allocation_value = .allocation_value;
3          curr_extension_value  = .extension_value;
3          curr_file_max_value   = .file_max_value;
3          curr_protection_or    = .protection_or;
3          curr_protection_and   = .protection_and;
3          curr_volume_value     = .volume_value;
2          END;
2
2
2      ! Determine if this is a COPY or APPEND command.
2      !
2      IF NOT .append_command
2      THEN
3          BEGIN
3
3          ! It is a COPY command.  Parse the COPY specific qualifiers.
3          !
3          ! Initialize the flags for the local qualifier states.  Assume that
3          ! there will be no local qualifier.  See if the qualifier is present.
3          ! If is is, see if it is locally present or locally negated.  Set the
3          ! appropriate flags.  The output file can not be open for the local
3          ! qualifiers to be accepted.  The local qualifiers effect the attributes
3          ! of the output file at creation time (i.e. allocation, location, etc.).
3          ! These things cannot change once the file is open.  Therefore, if the
3          ! output file is open and a local qualifier has been encountered, issue
3          ! a warning, ignore the qualifier and continue processing.
3          !
3          ! /OVERLAY
3          !
3          loc_overlay_qual = neg_overlay_qual = FALSE;
3          rtn_status = CLI$PRESENT( overlay_desc );
3          SELECTONE .rtn_status OF
3              SET
3              [CLI$_LOCPRES] : IF NOT .outfile_open
3                               THEN loc_overlay_qual = TRUE
3                               ELSE PUT_MESSAGE( MSG$_REPLACED, 1, MSG_DESC );
3
3              [CLI$_LOCNEG]  : IF NOT .outfile_open
3                               THEN neg_overlay_qual = TRUE
3                               ELSE PUT_MESSAGE( MSG$_REPLACED, 1, MSG_DESC );
3              TES;
3
3
3          ! /REPLACE
3          !
3          loc_replace_qual = neg_replace_qual = FALSE;
3          rtn_status = CLI$PRESENT( replace_desc );
3          SELECTONE .rtn_status OF
3              SET
3              [CLI$_LOCPRES] : IF NOT .outfile_open
3                               THEN loc_replace_qual = TRUE
3                               ELSE PUT_MESSAGE( MSG$_REPLACED, 1, MSG_DESC );
3
3              [CLI$_LOCNEG]  : IF NOT .outfile_open
3                               THEN neg_replace_qual = TRUE
3                               ELSE PUT_MESSAGE( MSG$_REPLACED, 1, MSG_DESC );
```

```
        TES;


    ! /TRUNCATE
    !
    loc_truncate_qual = neg_truncate_qual = FALSE;
    rtn_status = CLI$PRESENT( truncate_desc );
    SELECTONE .rtn_status OF
        SET
        [CLI$_LOCPRES] : IF NOT .outfile_open
                         THEN loc_truncate_qual = TRUE
                         ELSE PUT_MESSAGE( MSG$_REPLACED, 1, MSG_DESC );

        [CLI$_LOCNEG]  : IF NOT .outfile_open
                         THEN neg_truncate_qual = TRUE
                         ELSE PUT_MESSAGE( MSG$_REPLACED, 1, MSG_DESC );
        TES;


    ! /VOLUME
    !
    loc_volume_qual = neg_volume_qual = FALSE;
    rtn_status = CLI$PRESENT( volume_desc );
    SELECTONE .rtn_status OF
        SET
        [CLI$_LOCPRES] : IF NOT .outfile_open
                         THEN
                            BEGIN

                            ! Get the value and and convert it from a string into binary.
                            !
                            CLI$GET_VALUE( volume_desc, cli_desc );
                            IF NOT (rtn_status = [IB$CVT_DTB(.cli_desc[ DSC$W_LENGTH ],
                                                 .cli_desc[ DSC$A_POINTER ], curr_volume_value))
                            THEN
                                PUT_MESSAGEX( MSG$_INVQUAVAL, 2, cli_desc, volume_desc );
                            loc_volume_qual = TRUE;
                            END
                         ELSE PUT_MESSAGE( MSG$_REPLACED, 1, MSG_DESC );

        [CLI$_LOCNEG]  : IF NOT .outfile_open
                         THEN neg_volume_qual = TRUE
                         ELSE PUT_MESSAGE( MSG$_REPLACED, 1, MSG_DESC );
        TES;
    END;

! Parse the qualifiers which are applicable to both commands.
!
! /READ_CHECK, /WRITE_CHECK and /CONTIGUOUS
!
loc_read_chk_qual = neg_read_chk_qual = FALSE;
rtn_status = CLI$PRESENT( read_check_desc );
SELECTONE .rtn_status OF
    SET
    [CLI$_LOCPRES] : IF NOT .outfile_open
                     THEN loc_read_chk_qual = TRUE
                     ELSE PUT_MESSAGE( MSG$_REPLACED, 1, MSG_DESC );

    [CLI$_LOCNEG]  : IF NOT .outfile_open
                     THEN neg_read_chk_qual = TRUE
                     ELSE PUT_MESSAGE( MSG$_REPLACED, 1, MSG_DESC );
    TES;


loc_write_chk_qual = neg_write_chk_qual = FALSE;
rtn_status = CLI$PRESENT( write_check_desc );
SELECTONE .rtn_status OF
    SET
    [CLI$_LOCPRES] : IF NOT .outfile_open
                     THEN loc_write_chk_qual = TRUE
                     ELSE PUT_MESSAGE( MSG$_REPLACED, 1, MSG_DESC );

    [CLI$_LOCNEG]  : IF NOT .outfile_open
                     THEN neg_write_chk_qual = TRUE
                     ELSE PUT_MESSAGE( MSG$_REPLACED, 1, MSG_DESC );
    TES;


loc_contig_qual = neg_contig_qual = FALSE;
rtn_status = CLI$PRESENT( contiguous_desc );
SELECTONE .rtn_status OF
    SET
    [CLI$_LOCPRES] : IF NOT .outfile_open
                     THEN loc_contig_qual = TRUE
                     ELSE PUT_MESSAGE( MSG$_REPLACED, 1, MSG_DESC );

    [CLI$_LOCNEG]  : IF NOT .outfile_open
                     THEN neg_contig_qual = TRUE
                     ELSE PUT_MESSAGE( MSG$_REPLACED, 1, MSG_DESC );
    TES;


! /ALLOCATION
!
loc_alloc_qual = neg_alloc_qual = FALSE;
rtn_status = CLI$PRESENT( allocation_desc );
SELECTONE .rtn_status OF
    SET
    [CLI$_LOCPRES] : IF NOT .outfile_open
                     THEN
                        BEGIN

                        ! Get the value and and convert it from a string into binary.
                        !
                        CLI$GET_VALUE( allocation_desc, cli_desc );
                        IF NOT (rtn_status = LIB$CVT_DTB(.cli_desc[ DSC$W_LENGTH ],
                                                    .cli_desc[ DSC$A_POINTER ], curr_allocation_value))
                        THEN
                            PUT_MESSAGEX( MSG$_INVQUAVAL, 2, cli_desc, allocation_desc );
                        loc_alloc_qual = TRUE;
                        END
                     ELSE PUT_MESSAGE( MSG$_REPLACED, 1, MSG_DESC );

    [CLI$_LOCNEG]  : IF NOT .outfile_open
```

```
 2                          THEN neg_alloc_qual = TRUE
 2                          ELSE PUT_MESSAGE( MSG$_REPLACED, 1, MSG_DESC );
 2          TES;
 2
 2
 2      ! /EXTENSION
 2      !
 2      loc_extend_qual = neg_extend_qual = FALSE;
 2      rtn_status = CLI$PRESENT( extension_desc );
 2      SELECTONE .rtn_status OF
 2          SET
 2          [CLI$_LOCPRES] : IF NOT .outfile_open
 2                           THEN
 3                              BEGIN
 3
 3                              ! Get the value and and convert it from a string into binary.
 3                              !
 3                              CLI$GET_VALUE( extension_desc, cli_desc );
 4                              IF NOT (rtn_status = LIB$CVT_DTB(.cli_desc[ DSC$W_LENGTH ],
 4                                                  .cli_desc[ DSC$A_POINTER ], curr_extension_value))
 3                              THEN
 3                                  PUT_MESSAGEX( MSG$_INVQUAVAL, 2, cli_desc, extension_desc );
 3                              loc_extend_qual = TRUE;
 3                              END
 2                           ELSE PUT_MESSAGE( MSG$_REPLACED, 1, MSG_DESC );
 2
 2          [CLI$_LOCNEG]  : IF NOT .outfile_open
 2                           THEN neg_extend_qual = TRUE
 2                           ELSE PUT_MESSAGE( MSG$_REPLACED, 1, MSG_DESC );
 2          TES;
 2
 2
 2      ! /FILE_MAXIMUM
 2      !
 2      loc_file_max_qual = neg_file_max_qual = FALSE;
 2      rtn_status = CLI$PRESENT( file_max_desc );
 2      SELECTONE .rtn_status OF
 2          SET
 2          [CLI$_LOCPRES] : IF NOT .outfile_open
 2                           THEN
 3                              BEGIN
 3
 3                              ! Get the value and and convert it from a string into binary.
 3                              !
 3                              CLI$GET_VALUE( file_max_desc, cli_desc );
 4                              IF NOT (rtn_status = LIB$CVT_DTB(.cli_desc[ DSC$W_LENGTH ],
 4                                                  .cli_desc[ DSC$A_POINTER ], curr_file_max_value))
 3                              THEN
 3                                  PUT_MESSAGEX( MSG$_INVQUAVAL, 2, cli_desc, file_max_desc );
 3                              loc_file_max_qual = TRUE;
 3                              END
 2                           ELSE PUT_MESSAGE( MSG$_REPLACED, 1, MSG_DESC );
 2
 2          [CLI$_LOCNEG]  : IF NOT .outfile_open
 2                           THEN neg_extend_qual = TRUE
 2                           ELSE PUT_MESSAGE( MSG$_REPLACED, 1, MSG_DESC );
 2          TES;
```

```
2
2
2       ! /PROTECTION
2       !
2       loc_protect_qual = neg_protect_qual = FALSE;
2       rtn_status = CLI$PRESENT( protection_desc );
2       SELECTONE .rtn_status OF
2           SET
2           [CLI$_LOCPRES] : IF NOT .outfile_open
2                            THEN
3                               BEGIN
3                               !
3                               ! Parse the keyword values and save the results.
3                               !
3                               protection_parse();
3                               loc_protect_qual = TRUE;
3                               END
2                            ELSE PUT_MESSAGE( MSG$_REPLACED, 1, MSG_DESC );
2
2           [CLI$_LOCNEG]  : IF NOT .outfile_open
2                            THEN neg_protect_qual = TRUE
2                            ELSE PUT_MESSAGE( MSG$_REPLACED, 1, MSG_DESC );
2           TES;
2
1       END;                                                  ! routine COPY$GET_GLOBAL_QUAL


                                                                   .PSECT  $PLIT$,NOWRT,NOEXE,2

75 71 20 65 67 6E 61 68 63 20 74 22 6E 61 63  00110 P.ABF:  .ASCII  \can"t change quals in the middle of the \  ;
64 64 69 6D 20 65 68 74 20 6E 69 20 73 6C 61  0011F                                                           ;
               20 65 68 74 20 66 6F 20 65 6C  0012E                                                           ;
                           64 6E 61 6D 6D 6F 63  00138          .ASCII  \command\                              ;
                                              0013F          .BLKB   1
                                    0000002F  00140 P.ABE:  .LONG   47                                      ;
                                    00000000' 00144          .ADDRESS P.ABF                                 ;

                                                    MSG_DESC=           P.ABE


                                                             .PSECT  $CODE$,NOWRT,2

                                          0FFC 00000         .ENTRY  COPY$GET_LOCAL_QUAL, Save R2,R3,R4,R5,R6,-  ; 0926
                                                                     R7,R8,R9,R10,R11                            ;
                           5B 00000000G  8F  D0 00002        MOVL    #CLI$_LOCPRES, R11                          ;
                           5A     0000G  CF  9E 00009        MOVAB   COPY$SEM_STATUS, R10                        ;
                           59     0000G  CF  9E 0000E        MOVAB   COPY$MSG_NUMBER, R9                         ;
                           58 00000000G  00  9E 00013        MOVAB   LIB$SIGNAL, R8                              ;
                           57     0000'  CF  9E 0001A        MOVAB   MSG_DESC, R7                                ;
                           56     0000G  CF  9E 0001F        MOVAB   COPY$CLI_STATUS+4, R6                       ;
                           5E            08  C2 00024        SUBL2   #8, SP                                      ;
      08          00       6E            00  2C 00027        MOVC5   #0, (SP), #0, #8, CLI_DESC                  ; 0978
                                         6E     0002C                                                            ;
                     03    AE            02  90 0002D        MOVB    #2, CLI_DESC+3                              ; 0979
              1E     02    AA            01  E0 00031        BBS     #1, COPY$SEM_STATUS+2, 1$                   ; 0981
                  0000'    CF      04    A6  7D 00036        MOVQ    COPY$CLI_STATUS+8, CURR_ALLOCATION_VALUE    ; 0984
```

```
                 0000'  CF        0C    A6  D0 0003C          MOVL    COPY$CLI_STATUS+16, CURR_FILE_MAX_VALUE    ; 0986
                 0000'  CF        10    A6  B0 00042          MOVW    COPY$CLI_STATUS+20, CURR_PROTECTION_OR     ; 0987
                 0000'  CF        12    A6  B0 00048          MOVW    COPY$CLI_STATUS+22, CURR_PROTECTION_AND    ; 0988
                 0000'  CF        14    A6  D0 0004E          MOVL    COPY$CLI_STATUS+24, CURR_VOLUME_VALUE      ; 0989
                        03        FC    A6  E9 00054 1$:      BLBC    COPY$CLI_STATUS, 2$                        ; 0995
                                      017C  31 00058          BRW     17$                                        ;
                    01  A6              03  8A 0005B 2$:      BICB2   #3, COPY$CLI_STATUS+5                      ; 1013
                                  98    A7  9F 0005F          PUSHAB  OVERLAY_DESC                               ; 1014
                 0000G  CF              01  FB 00062          CALLS   #1, CLI$PRESENT                            ;
                        52              50  D0 00067          MOVL    R0, RTN_STATUS                             ;
                        5B              52  D1 0006A          CMPL    RTN_STATUS, R11                            ; 1017
                                        08  12 0006D          BNEQ    3$                                         ;
         1A         02  AA              01  E0 0006F          BBS     #1, COPY$SEM_STATUS+2, 4$                  ;
                    01  A6              01  88 00074          BISB2   #1, COPY$CLI_STATUS+5                      ; 1018
                                        25  11 00078          BRB     5$                                         ;
             00000000G  8F              52  D1 0007A 3$:      CMPL    RTN_STATUS, #CLI$_LOCNEG                   ; 1021
                                        1C  12 00081          BNEQ    5$                                         ;
         06         02  AA              01  E0 00083          BBS     #1, COPY$SEM_STATUS+2, 4$                  ;
                    01  A6              02  88 00088          BISB2   #2, COPY$CLI_STATUS+5                      ; 1022
                                        11  11 0008C          BRB     5$                                         ;
                                        57  DD 0008E 4$:      PUSHL   R7                                         ; 1023
                                        01  DD 00090          PUSHL   #1                                         ;
                        7E      10BB    8F  3C 00092          MOVZWL  #4283, -(SP)                               ;
                        69              01  FB 00097          CALLS   #1, COPY$MSG_NUMBER                        ;
                                        50  DD 0009A          PUSHL   R0                                         ;
                        68              03  FB 0009C          CALLS   #3, LIB$SIGNAL                             ;
                    02  A6              0C  8A 0009F 5$:      BICB2   #12, COPY$CLI_STATUS+6                     ; 1029
                                  C8    A7  9F 000A3          PUSHAB  REPLACE_DESC                               ; 1030
                 0000G  CF              01  FB 000A6          CALLS   #1, CLI$PRESENT                            ;
                        52              50  D0 000AB          MOVL    R0, RTN_STATUS                             ;
                        5B              52  D1 000AE          CMPL    RTN_STATUS, R11                            ; 1033
                                        0B  12 000B1          BNEQ    6$                                         ;
         1A         02  AA              01  E0 000B3          BBS     #1, COPY$SEM_STATUS+2, 7$                  ;
                    02  A6              04  88 000B8          BISB2   #4, COPY$CLI_STATUS+6                      ; 1034
                                        25  11 000BC          BRB     8$                                         ;
             00000000G  8F              52  D1 000BE 6$:      CMPL    RTN_STATUS, #CLI$_LOCNEG                   ; 1037
                                        1C  12 000C5          BNEQ    8$                                         ;
         06         02  AA              01  E0 000C7          BBS     #1, COPY$SEM_STATUS+2, 7$                  ;
                    02  A6              08  88 000CC          BISB2   #8, COPY$CLI_STATUS+6                      ; 1038
                                        11  11 000D0          BRB     8$                                         ;
                                        57  DD 000D2 7$:      PUSHL   R7                                         ; 1039
                                        01  DD 000D4          PUSHL   #1                                         ;
                        7E      10BB    8F  3C 000D6          MOVZWL  #4283, -(SP)                               ;
                        69              01  FB 000DB          CALLS   #1, COPY$MSG_NUMBER                        ;
                                        50  DD 000DE          PUSHL   R0                                         ;
                        68              03  FB 000E0          CALLS   #3, LIB$SIGNAL                             ;
                    01  A6      0180    8F  AA 000E3 8$:      BICW2   #384, COPY$CLI_STATUS+5                    ; 1045
                                  B8    A7  9F 000E9          PUSHAB  TRUNCATE_DESC                              ; 1046
                 0000G  CF              01  FB 000EC          CALLS   #1, CLI$PRESENT                            ;
                        52              50  D0 000F1          MOVL    R0, RTN_STATUS                             ;
                        5B              52  D1 000F4          CMPL    RTN_STATUS, R11                            ; 1049
                                        0C  12 000F7          BNEQ    9$                                         ;
         1B         02  AA              01  E0 000F9          BBS     #1, COPY$SEM_STATUS+2, 10$                 ;
                    01  A6        80    8F  88 000FE          BISB2   #128, COPY$CLI_STATUS+5                    ; 1050
                                        25  11 00103          BRB     11$                                        ;
             00000000G  8F              52  D1 00105 9$:      CMPL    RTN_STATUS, #CLI$_LOCNEG                   ; 1053
                                        1C  12 0010C          BNEQ    11$                                        ;
```

```
        06             02  AA            01 E0 0010E         BBS     #1, COPY$SEM_STATUS+2, 10$               ;
                       02  A6            01 88 00113         BISB2   #1, COPY$CLI_STATUS+6                    ; 1054
                                         11 11 00117         BRB     11$                                      ;
                                         57 DD 00119 10$:    PUSHL   R7                                       ; 1055
                                         01 DD 0011B         PUSHL   #1                                       ;
                           7E     10BB   8F 3C 0011D         MOVZWL  #4283, -(SP)                             ;
                           69            01 FB 00122         CALLS   #1, COPY$MSG_NUMBER                      ;
                                         50 DD 00125         PUSHL   R0                                       ;
                           68            03 FB 00127         CALLS   #3, LIB$SIGNAL                           ;
                       01  A6            18 8A 0012A 11$:    BICB2   #24, COPY$CLI_STATUS+5                   ; 1061
                                    A8   A7 9F 0012E         PUSHAB  VOLUME_DESC                              ; 1062
                    0000G  CF            01 FB 00131         CALLS   #1, CLI$PRESENT                          ;
                           52            50 D0 00136         MOVL    R0, RTN_STATUS                           ;
                           5B            52 D1 00139         CMPL    RTN_STATUS, R11                          ; 1065
                                         74 12 0013C         BNEQ    14$                                      ;
        78             02  AA            01 E0 0013E         BBS     #1, COPY$SEM_STATUS+2, 15$               ;
                                         5E DD 00143         PUSHL   SP                                       ; 1071
                                    A8   A7 9F 00145         PUSHAB  VOLUME_DESC                              ;
                    0000G  CF            02 FB 00148         CALLS   #2, CLI$GET_VALUE                        ;
                                 0000'   CF 9F 0014D         PUSHAB  CURR_VOLUME_VALUE                        ; 1072
                                    08   AE DD 00151         PUSHL   CLI_DESC+4                               ; 1073
                           7E       08   AE 3C 00154         MOVZWL  CLI_DESC, -(SP)                          ; 1072
                00000000G  00            03 FB 00158         CALLS   #3, LIB$CVT_DTB                          ;
                           52            50 D0 0015F         MOVL    R0, RTN_STATUS                           ;
                           47            52 E8 00162         BLBS    RTN_STATUS, 13$                          ;
                           7E     132C   8F 3C 00165         MOVZWL  #4908, -(SP)                             ; 1075
                           69            01 FB 0016A         CALLS   #1, COPY$MSG_NUMBER                      ;
7E      00                 50            01 7A 0016D         EMUL    #1, R0, #0, -(SP)                        ;
50      50                 8E            08 7B 00172         EDIV    #8, (SP)+, R0, R0                        ;
                           04            50 D1 00177         CMPL    R0, #4                                   ;
                                         17 13 0017A         BEQL    12$                                      ;
                                    A8   A7 9F 0017C         PUSHAB  VOLUME_DESC                              ;
                                    04   AE 9F 0017F         PUSHAB  CLI_DESC                                 ;
                                         02 DD 00182         PUSHL   #2                                       ;
                           7E     132C   8F 3C 00184         MOVZWL  #4908, -(SP)                             ;
                           69            01 FB 00189         CALLS   #1, COPY$MSG_NUMBER                      ;
                                         50 DD 0018C         PUSHL   R0                                       ;
                           68            04 FB 0018E         CALLS   #4, LIB$SIGNAL                           ;
                                         19 11 00191         BRB     13$                                      ;
                                    A8   A7 9F 00193 12$:    PUSHAB  VOLUME_DESC                              ;
                                    04   AE 9F 00196         PUSHAB  CLI_DESC                                 ;
                                         02 DD 00199         PUSHL   #2                                       ;
                           7E     132C   8F 3C 0019B         MOVZWL  #4908, -(SP)                             ;
                           69            01 FB 001A0         CALLS   #1, COPY$MSG_NUMBER                      ;
                                         50 DD 001A3         PUSHL   R0                                       ;
                00000000G  00            04 FB 001A5         CALLS   #4, LIB$STOP                             ;
                       01  A6            08 88 001AC 13$:    BISB2   #8, COPY$CLI_STATUS+5                    ; 1076
                                         25 11 001B0         BRB     17$                                      ; 1065
                00000000G  8F            52 D1 001B2 14$:    CMPL    RTN_STATUS, #CLI$_LOCNEG                 ; 1080
                                         1C 12 001B9         BNEQ    17$                                      ;
        06             02  AA            01 E0 001BB 15$:    BBS     #1, COPY$SEM_STATUS+2, 16$               ;
                       01  A6            10 88 001C0         BISB2   #16, COPY$CLI_STATUS+5                   ; 1081
                                         11 11 001C4         BRB     17$                                      ;
                                         57 DD 001C6 16$:    PUSHL   R7                                       ; 1082
                                         01 DD 001C8         PUSHL   #1                                       ;
                           7E     10BB   8F 3C 001CA         MOVZWL  #4283, -(SP)                             ;
                           69            01 FB 001CF         CALLS   #1, COPY$MSG_NUMBER                      ;
```

```
                                 50  DD 001D2         PUSHL   R0                                      ;
                  68             03  FB 001D4         CALLS   #3, LIB$SIGNAL                          ;
                  66             06  8A 001D7 17$:    BICB2   #6, COPY$CLI_STATUS+4                   ; 1090
                          FF74   C7  9F 001DA         PUSHAB  READ_CHECK_DESC                         ; 1091
            0000G CF             01  FB 001DE         CALLS   #1, CLI$PRESENT                         ;
                  52             50  D0 001E3         MOVL    R0, RTN_STATUS                          ;
                  5B             52  D1 001E6         CMPL    RTN_STATUS, R11                         ; 1094
                                 0A  12 001E9         BNEQ    18$                                     ;
18             02 AA             01  E0 001EB         BBS     #1, COPY$SEM_STATUS+2, 19$              ;
                  66             02  88 001F0         BISB2   #2, COPY$CLI_STATUS+4                   ; 1095
                                 24  11 001F3         BRB     20$                                     ;
       000000000G 8F             52  D1 001F5 18$:    CMPL    RTN_STATUS, #CLI$_LOCNEG                ; 1098
                                 1B  12 001FC         BNEQ    20$                                     ;
05             02 AA             01  E0 001FE         BBS     #1, COPY$SEM_STATUS+2, 19$              ;
                  66             04  88 00203         BISB2   #4, COPY$CLI_STATUS+4                   ; 1099
                                 11  11 00206         BRB     20$                                     ;
                                 57  DD 00208 19$:    PUSHL   R7                                      ; 1100
                                 01  DD 0020A         PUSHL   #1                                      ;
                  7E      10BB   8F  3C 0020C         MOVZWL  #4283, -(SP)                            ;
                  69             01  FB 00211         CALLS   #1, COPY$MSG_NUMBER                     ;
                                 50  DD 00214         PUSHL   R0                                      ;
                  68             03  FB 00216         CALLS   #3, LIB$SIGNAL                          ;
                  66        60   8F  8A 00219 20$:    BICB2   #96, COPY$CLI_STATUS+4                  ; 1104
                            88   A7  9F 0021D         PUSHAB  WRITE_CHECK_DESC                        ; 1105
            0000G CF             01  FB 00220         CALLS   #1, CLI$PRESENT                         ;
                  52             50  D0 00225         MOVL    R0, RTN_STATUS                          ;
                  5B             52  D1 00228         CMPL    RTN_STATUS, R11                         ; 1108
                                 0A  12 0022B         BNEQ    21$                                     ;
19             02 AA             01  E0 0022D         BBS     #1, COPY$SEM_STATUS+2, 22$              ;
                  66             20  88 00232         BISB2   #32, COPY$CLI_STATUS+4                  ; 1109
                                 25  11 00235         BRB     23$                                     ;
       000000000G 8F             52  D1 00237 21$:    CMPL    RTN_STATUS, #CLI$_LOCNEG                ; 1112
                                 1C  12 0023E         BNEQ    23$                                     ;
06             02 AA             01  E0 00240         BBS     #1, COPY$SEM_STATUS+2, 22$              ;
                  66        40   8F  88 00245         BISB2   #64, COPY$CLI_STATUS+4                  ; 1113
                                 11  11 00249         BRB     23$                                     ;
                                 57  DD 0024B 22$:    PUSHL   R7                                      ; 1114
                                 01  DD 0024D         PUSHL   #1                                      ;
                  7E      10BB   8F  3C 0024F         MOVZWL  #4283, -(SP)                            ;
                  69             01  FB 00254         CALLS   #1, COPY$MSG_NUMBER                     ;
                                 50  DD 00257         PUSHL   R0                                      ;
                  68             03  FB 00259         CALLS   #3, LIB$SIGNAL                          ;
               FE A6        60   8F  8A 0025C 23$:    BICB2   #96, COPY$CLI_STATUS+2                  ; 1118
                          FF24   C7  9F 00261         PUSHAB  CONTIGUOUS_DESC                         ; 1119
            0000G CF             01  FB 00265         CALLS   #1, CLI$PRESENT                         ;
                  52             50  D0 0026A         MOVL    R0, RTN_STATUS                          ;
                  5B             52  D1 0026D         CMPL    RTN_STATUS, R11                         ; 1122
                                 0B  12 00270         BNEQ    24$                                     ;
1B             02 AA             01  E0 00272         BBS     #1, COPY$SEM_STATUS+2, 25$              ;
               FE A6             20  88 00277         BISB2   #32, COPY$CLI_STATUS+2                  ; 1123
                                 26  11 0027B         BRB     26$                                     ;
       000000000G 8F             52  D1 0027D 24$:    CMPL    RTN_STATUS, #CLI$_LOCNEG                ; 1126
                                 1D  12 00284         BNEQ    26$                                     ;
07             02 AA             01  E0 00286         BBS     #1, COPY$SEM_STATUS+2, 25$              ;
               FE A6        40   8F  88 0028B         BISB2   #64, COPY$CLI_STATUS+2                  ; 1127
                                 11  11 00290         BRB     26$                                     ;
                                 57  DD 00292 25$:    PUSHL   R7                                      ; 1128
```

```
                                                 01  DD 00294           PUSHL   #1                             ;
                                   7E    10BB  8F  3C 00296           MOVZWL  #4283, -(SP)                   ;
                                   69          01  FB 0029B           CALLS   #1, COPY$MSG_NUMBER            ;
                                               50  DD 0029E           PUSHL   R0                             ;
                                   68          03  FB 002A0           CALLS   #3, LIB$SIGNAL                 ;
                                FE A6          06  8A 002A3 26$:      BICB2   #6, COPY$CLI_STATUS+2          ; 1134
                                         FF10  C7  9F 002A7           PUSHAB  ALLOCATION_DESC                ; 1135
                             0000G CF          01  FB 002AB           CALLS   #1, CLI$PRESENT                ;
                                   52          50  D0 002B0           MOVL    R0, RTN_STATUS                 ;
                                   5B          52  D1 002B3           CMPL    RTN_STATUS, R11                ; 1138
                                               77  12 002B6           BNEQ    29$                            ;
                    7B          02 AA          01  E0 002B8           BBS     #1, COPY$SEM_STATUS+2, 30$     ;
                                               5E  DD 002BD           PUSHL   SP                             ; 1144
                                         FF10  C7  9F 002BF           PUSHAB  ALLOCATION_DESC                ;
                             0000G CF          02  FB 002C3           CALLS   #2, CLI$GET_VALUE              ;
                                        0000'  CF  9F 002C8           PUSHAB  CURR_ALLOCATION_VALUE          ; 1145
                                           08  AE  DD 002CC           PUSHL   CLI_DESC+4                     ; 1146
                                   7E      08  AE  3C 002CF           MOVZWL  CLI_DESC, -(SP)                ; 1145
                        000000000G 00          03  FB 002D3           CALLS   #3, LIB$CVT_DTB                ;
                                   52          50  D0 002DA           MOVL    R0, RTN_STATUS                 ;
                                   49          52  E8 002DD           BLBS    RTN_STATUS, 28$                ;
                                   7E    132C  8F  3C 002E0           MOVZWL  #4908, -(SP)                   ; 1148
                                   69          01  FB 002E5           CALLS   #1, COPY$MSG_NUMBER            ;
        7E          00             50          01  7A 002E8           EMUL    #1, R0, #0, -(SP)              ;
        50          50             8E          08  7B 002ED           EDIV    #8, (SP)+, R0, R0              ;
                                   04          50  D1 002F2           CMPL    R0, #4                         ;
                                               18  13 002F5           BEQL    27$                            ;
                                         FF10  C7  9F 002F7           PUSHAB  ALLOCATION_DESC                ;
                                           04  AE  9F 002FB           PUSHAB  CLI_DESC                       ;
                                               02  DD 002FE           PUSHL   #2                             ;
                                   7E    132C  8F  3C 00300           MOVZWL  #4908, -(SP)                   ;
                                   69          01  FB 00305           CALLS   #1, COPY$MSG_NUMBER            ;
                                               50  DD 00308           PUSHL   R0                             ;
                                   68          04  FB 0030A           CALLS   #4, LIB$SIGNAL                 ;
                                               1A  11 0030D           BRB     28$                            ;
                                         FF10  C7  9F 0030F 27$:      PUSHAB  ALLOCATION_DESC                ;
                                           04  AE  9F 00313           PUSHAB  CLI_DESC                       ;
                                               02  DD 00316           PUSHL   #2                             ;
                                   7E    132C  8F  3C 00318           MOVZWL  #4908, -(SP)                   ;
                                   69          01  FB 0031D           CALLS   #1, COPY$MSG_NUMBER            ;
                                               50  DD 00320           PUSHL   R0                             ;
                        000000000G 00          04  FB 00322           CALLS   #4, LIB$STOP                   ;
                                FE A6          02  88 00329 28$:      BISB2   #2, COPY$CLI_STATUS+2          ; 1149
                                               25  11 0032D           BRB     32$                            ; 1138
                        000000000G 8F          52  D1 0032F 29$:      CMPL    RTN_STATUS, #CLI$_LOCNEG       ; 1153
                                               1C  12 00336           BNEQ    32$                            ;
                    06          02 AA          01  E0 00338 30$:      BBS     #1, COPY$SEM_STATUS+2, 31$     ;
                                FE A6          04  88 0033D           BISB2   #4, COPY$CLI_STATUS+2          ; 1154
                                               11  11 00341           BRB     32$                            ;
                                               57  DD 00343 31$:      PUSHL   R7                             ; 1155
                                               01  DD 00345           PUSHL   #1                             ;
                                   7E    10BB  8F  3C 00347           MOVZWL  #4283, -(SP)                   ;
                                   69          01  FB 0034C           CALLS   #1, COPY$MSG_NUMBER            ;
                                               50  DD 0034F           PUSHL   R0                             ;
                                   68          03  FB 00351           CALLS   #3, LIB$SIGNAL                 ;
                                FF A6          03  8A 00354 32$:      BICB2   #3, COPY$CLI_STATUS+3          ; 1161
                                         FF38  C7  9F 00358           PUSHAB  EXTENSION_DESC                 ; 1162
```

```
                          0000G  CF          01  FB 0035C         CALLS   #1, CLI$PRESENT                          ;
                                 52          50  D0 00361         MOVL    R0, RTN_STATUS                           ;
                                 5B          52  D1 00364         CMPL    RTN_STATUS, R11                          ; 1165
                                 77          12     00367         BNEQ    35$                                      ;
                   7B       02   AA          01  E0 00369         BBS     #1, COPY$SEM_STATUS+2, 36$               ;
                                             5E  DD 0036E         PUSHL   SP                                       ; 1171
                                     FF38    C7  9F 00370         PUSHAB  EXTENSION_DESC                           ;
                          0000G  CF          02  FB 00374         CALLS   #2, CLI$GET_VALUE                        ;
                                     0000'   CF  9F 00379         PUSHAB  CURR_EXTENSION_VALUE                     ; 1172
                                       08    AE  DD 0037D         PUSHL   CLI_DESC+4                               ; 1173
                                 7E    08    AE  3C 00380         MOVZWL  CLI_DESC, -(SP)                          ; 1172
                      00000000G  00          03  FB 00384         CALLS   #3, LIB$CVT_DTB                          ;
                                 52          50  D0 0038B         MOVL    R0, RTN_STATUS                           ;
                                 49          52  E8 0038E         BLBS    RTN_STATUS, 34$                          ;
                                 7E  132C    8F  3C 00391         MOVZWL  #4908, -(SP)                             ; 1175
                                 69          01  FB 00396         CALLS   #1, COPY$MSG_NUMBER                      ;
        7E         00            50          01  7A 00399         EMUL    #1, R0, #0, =(SP)                        ;
        50         50            8E          08  7B 0039E         EDIV    #8, (SP)+, R0, R0                        ;
                                 04          50  D1 003A3         CMPL    R0, #4                                   ;
                                             18  13 003A6         BEQL    33$                                      ;
                                     FF38    C7  9F 003A8         PUSHAB  EXTENSION_DESC                           ;
                                       04    AE  9F 003AC         PUSHAB  CLI_DESC                                 ;
                                             02  DD 003AF         PUSHL   #2                                       ;
                                 7E  132C    8F  3C 003B1         MOVZWL  #4908, -(SP)                             ;
                                 69          01  FB 003B6         CALLS   #1, COPY$MSG_NUMBER                      ;
                                             50  DD 003B9         PUSHL   R0                                       ;
                                 68          04  FB 003BB         CALLS   #4, LIB$SIGNAL                           ;
                                             1A  11 003BE         BRB     34$                                      ;
                                     FF38    C7  9F 003C0  33$:   PUSHAB  EXTENSION_DESC                           ;
                                       04    AE  9F 003C4         PUSHAB  CLI_DESC                                 ;
                                             02  DD 003C7         PUSHL   #2                                       ;
                                 7E  132C    8F  3C 003C9         MOVZWL  #4908, -(SP)                             ;
                                 69          01  FB 003CE         CALLS   #1, COPY$MSG_NUMBER                      ;
                                             50  DD 003D1         PUSHL   R0                                       ;
                      00000000G  00          04  FB 003D3         CALLS   #4, LIB$STOP                             ;
                             FF  A6          01  88 003DA  34$:   BISB2   #1, COPY$CLI_STATUS+3                    ; 1176
                                             25  11 003DE         BRB     38$                                      ; 1165
                      00000000G  8F          52  D1 003E0  35$:   CMPL    RTN_STATUS, #CLI$_LOCNEG                 ; 1180
                                             1C  12 003E7         BNEQ    38$                                      ;
                   06       02   AA          01  E0 003E9  36$:   BBS     #1, COPY$SEM_STATUS+2, 37$               ;
                            FF   A6          02  88 003EE         BISB2   #2, COPY$CLI_STATUS+3                    ; 1181
                                             11  11 003F2         BRB     38$                                      ;
                                             57  DD 003F4  37$:   PUSHL   R7                                       ; 1182
                                             01  DD 003F6         PUSHL   #1                                       ;
                                 7E  10BB    8F  3C 003F8         MOVZWL  #4283, -(SP)                             ;
                                 69          01  FB 003FD         CALLS   #1, COPY$MSG_NUMBER                      ;
                                             50  DD 00400         PUSHL   R0                                       ;
                                 68          03  FB 00402         CALLS   #3, LIB$SIGNAL                           ;
                            FF   A6          18  8A 00405  38$:   BICB2   #24, COPY$CLI_STATUS+3                   ; 1188
                                     FF4C    C7  9F 00409         PUSHAB  FILE_MAX_DESC                            ; 1189
                          0000G  CF          01  FB 0040D         CALLS   #1, CLI$PRESENT                          ;
                                 52          50  D0 00412         MOVL    R0, RTN_STATUS                           ;
                                 5B          52  D1 00415         CMPL    RTN_STATUS, R11                          ; 1192
                                             77  12 00418         BNEQ    41$                                      ;
                   7B       02   AA          01  E0 0041A         BBS     #1, COPY$SEM_STATUS+2, 42$               ;
                                             5E  DD 0041F         PUSHL   SP                                       ; 1198
                                     FF4C    C7  9F 00421         PUSHAB  FILE_MAX_DESC                            ;
```

```
                  0000G  CF         02  FB 00425          CALLS   #2, CLI$GET_VALUE
                              0000' CF  9F 0042A          PUSHAB  CURR_FILE_MAX_VALUE                        1199
                                 08 AE  DD 0042E          PUSHL   CLI_DESC+4                                 1200
                     7E          08 AE  3C 00431          MOVZWL  CLI_DESC, -(SP)                            1199
              00000000G  00         03  FB 00435          CALLS   #3, LIB$CVT_DTB
                         52         50  D0 0043C          MOVL    R0, RTN_STATUS
                         49         52  E8 0043F          BLBS    RTN_STATUS, 40$
                     7E        132C 8F  3C 00442          MOVZWL  #4908, -(SP)                               1202
                         69         01  FB 00447          CALLS   #1, COPY$MSG_NUMBER
   7E           00       50         01  7A 0044A          EMUL    #1, R0, #0, -(SP)
   50           50       8E         08  7B 0044F          EDIV    #8, (SP)+, R0, R0
                         04         50  D1 00454          CMPL    R0, #4
                                    18  13 00457          BEQL    39$
                               FF4C C7  9F 00459          PUSHAB  FILE_MAX_DESC
                                 04 AE  9F 0045D          PUSHAB  CLI_DESC
                                    02  DD 00460          PUSHL   #2
                     7E        132C 8F  3C 00462          MOVZWL  #4908, -(SP)
                         69         01  FB 00467          CALLS   #1, COPY$MSG_NUMBER
                                    50  DD 0046A          PUSHL   R0
                         68         04  FB 0046C          CALLS   #4, LIB$SIGNAL
                                    1A  11 0046F          BRB     40$
                               FF4C C7  9F 00471 39$:     PUSHAB  FILE_MAX_DESC
                                 04 AE  9F 00475          PUSHAB  CLI_DESC
                                    02  DD 00478          PUSHL   #2
                     7E        132C 8F  3C 0047A          MOVZWL  #4908, -(SP)
                         69         01  FB 0047F          CALLS   #1, COPY$MSG_NUMBER
                                    50  DD 00482          PUSHL   R0
              00000000G  00         04  FB 00484          CALLS   #4, LIB$STOP
                     FF  A6         08  88 0048B 40$:     BISB2   #8, COPY$CLI_STATUS+3                      1203
                                    25  11 0048F          BRB     44$                                        1192
              00000000G  8F         52  D1 00491 41$:     CMPL    RTN_STATUS, #CLI$_LOCNEG                   1207
                                    1C  12 00498          BNEQ    44$
            06       02  AA         01  E0 0049A 42$:     BBS     #1, COPY$SEM_STATUS+2, 43$
                     FF  A6         02  88 0049F          BISB2   #2, COPY$CLI_STATUS+3                      1208
                                    11  11 004A3          BRB     44$
                                    57  DD 004A5 43$:     PUSHL   R7                                         1209
                                    01  DD 004A7          PUSHL   #1
                     7E        10BB 8F  3C 004A9          MOVZWL  #4283, -(SP)
                         69         01  FB 004AE          CALLS   #1, COPY$MSG_NUMBER
                                    50  DD 004B1          PUSHL   R0
                         68         03  FB 004B3          CALLS   #3, LIB$SIGNAL
                     FF  A6      C0 8F  8A 004B6 44$:     BICB2   #192, COPY$CLI_STATUS+3                    1215
                               FF60 C7  9F 004BB          PUSHAB  PROTECTION_DESC                            1216
                  0000G  CF         01  FB 004BF          CALLS   #1, CLI$PRESENT
                         52         50  D0 004C4          MOVL    R0, RTN_STATUS
                         5B         52  D1 004C7          CMPL    RTN_STATUS, R11                            1219
                                    10  12 004CA          BNEQ    45$
            1F       02  AA         01  E0 004CC          BBS     #1, COPY$SEM_STATUS+2, 46$
                  0000V  CF         00  FB 004D1          CALLS   #0, PROTECTION_PARSE                       1225
                     FF  A6      40 8F  88 004D6          BISB2   #64, COPY$CLI_STATUS+3                     1226
                                        04 004DB          RET                                                1219
              00000000G  8F         52  D1 004DC 45$:     CMPL    RTN_STATUS, #CLI$_LOCNEG                   1230
                                    1C  12 004E3          BNEQ    47$
            06       02  AA         01  E0 004E5          BBS     #1, COPY$SEM_STATUS+2, 46$
                     FF  A6      80 8F  88 004EA          BISB2   #128, COPY$CLI_STATUS+3                    1231
                                        04 004EF          RET
                                    57  DD 004F0 46$:     PUSHL   R7                                         1232
```

```
                                         01  DD 004F2          PUSHL   #1                                   ;
                         7E      10BB    8F  3C 004F4          MOVZWL  #4283, -(SP)                         ;
                         69              01  FB 004F9          CALLS   #1, COPY$MSG_NUMBER                  ;
                                         50  DD 004FC          PUSHL   R0                                   ;
                         68              03  FB 004FE          CALLS   #3, LIB$SIGNAL                       ;
                                             04 00501 47$:     RET                                          ; 1235

; Routine Size:  1282 bytes,    Routine Base:  $CODE$ + 0386
```

```
1 ROUTINE PROTECTION_PARSE : NOVALUE =
1
1 !++
1 ! FUNCTIONAL DESCRIPTION:
1 !
1 !       This routine parses a PROTECTION qualifier value.
1 !
1 ! FORMAL PARAMETERS:
1 !
1 !       NONE
1 !
1 ! IMPLICIT INPUTS:
1 !
1 !       NONE
1 !
1 ! IMPLICIT OUTPUTS:
1 !
1 !       CURR_PROTECTION_OR  - Protection mask storage
1 !       CURR_PROTECTION_AND - Protection mask storage
1 !
1 ! ROUTINE VALUE:
1 !
1 !       None
1 !
1 ! SIDE EFFECTS:
1 !
1 !       None
1 !
1 !--
1
2     BEGIN
2
2     MAP
2         CURR_PROTECTION_OR  :  $BBLOCK[ 2 ],            ! Protection mask
2         CURR_PROTECTION_AND :  $BBLOCK[ 2 ];            ! Protection mask
2
2     LOCAL
2         RTN_STATUS,                                     ! Keyword lookup completion code
2         CLI_DESC :                                      ! Descriptor which points to the keyword value
2             $BBLOCK[ DSC$C_S_BLN ],
2         KEY_DISP;                                       ! Displacement of keyword nibble in XAB$W_PRO
2
2     BIND
2         SYSTEM_DESC = $DESCRIPTOR('SYSTEM'),
2         OWNER_DESC  = $DESCRIPTOR('OWNER'),
2         GROUP_DESC  = $DESCRIPTOR('GROUP'),
2         WORLD_DESC  = $DESCRIPTOR('WORLD');
2
2
2     !
2     ! Initialize descriptor.
2     !
2     CH$FILL( 0, DSC$C_S_BLN, cli_desc);
2     cli_desc[ DSC$B_CLASS ] = DSC$K_CLASS_D;
2
2     !
2     ! Check for SYSTEM keyword.
```

```
2       !
2       IF CLI$PRESENT( SYSTEM_DESC )
2       THEN
3           BEGIN
3
3           ! Note that this is SYSTEM access.  Initialize the the system protection
3           ! fields, in case noaccess is specified.
3           !
3           KEY_DISP = BIT_LOCATION( XAB$V_SYS );
3           CURR_PROTECTION_OR[ prot_mask( .KEY_DISP, 4)]  = -1;
3           CURR_PROTECTION_AND[ prot_mask( .KEY_DISP, 4)]  = 0;
3
3           ! Retrieve the keyword value, if any, and parse it.
3           !
3           IF CLI$GET_VALUE( SYSTEM_DESC, CLI_DESC )
3           THEN
3               PARSE_PROTECTION_VALUE( CLI_DESC, .KEY_DISP );
3
2           END;                                             ! SYSTEM parse
2
2
2       ! Check for OWNER keyword.
2       !
2       IF CLI$PRESENT( OWNER_DESC )
2       THEN
3           BEGIN
3
3           ! Note that this is OWNER access.  Initialize the the OWNER protection
3           ! fields, in case noaccess is specified.
3           !
3           KEY_DISP = BIT_LOCATION( XAB$V_OWN );
3           CURR_PROTECTION_OR[ prot_mask( .KEY_DISP, 4)]  = -1;
3           CURR_PROTECTION_AND[ prot_mask( .KEY_DISP, 4)]  = 0;
3
3           ! Retrieve the keyword value, if any, and parse it.
3           !
3           IF CLI$GET_VALUE( OWNER_DESC, CLI_DESC )
3           THEN
3               PARSE_PROTECTION_VALUE( CLI_DESC, .KEY_DISP );
3
2           END;                                             ! OWNER parse
2
2
2       ! Check for GROUP keyword.
2       !
2       IF CLI$PRESENT( GROUP_DESC )
2       THEN
3           BEGIN
3
3           ! Note that this is GROUP access.  Initialize the the GROUP protection
3           ! fields, in case noaccess is specified.
3           !
3           KEY_DISP = BIT_LOCATION( XAB$V_GRP );
3           CURR_PROTECTION_OR[ prot_mask( .KEY_DISP, 4)]  = -1;
3           CURR_PROTECTION_AND[ prot_mask( .KEY_DISP, 4)]  = 0;
3
3           ! Retrieve the keyword value, if any, and parse it.
```

```
;   3           !
;   3           IF CLI$GET_VALUE( GROUP_DESC, CLI_DESC )
;   3           THEN
;   3               PARSE_PROTECTION_VALUE( CLI_DESC, .KEY_DISP );
;   3
;   2           END;                                            ! GROUP parse
;   2
;   2
;   2       ! Check for WORLD keyword.
;   2       !
;   2       IF CLI$PRESENT( WORLD_DESC )
;   2       THEN
;   3           BEGIN
;   3
;   3           ! Note that this is WORLD access.  Initialize the the WORLD protection
;   3           ! fields, in case noaccess is specified.
;   3           !
;   3           KEY_DISP = BIT_LOCATION( XAB$V_WLD );
;   3           CURR_PROTECTION_OR[ prot_mask( .KEY_DISP, 4)]  = -1;
;   3           CURR_PROTECTION_AND[ prot_mask( .KEY_DISP, 4)]  = 0;
;   3
;   3           ! Retrieve the keyword value, if any, and parse it.
;   3           !
;   3           IF CLI$GET_VALUE( WORLD_DESC, CLI_DESC )
;   3           THEN
;   3               PARSE_PROTECTION_VALUE( CLI_DESC, .KEY_DISP );
;   3
;   2           END;                                            ! WORLD parse
;   2
;   2       RETURN;                                             ! Return to the caller.
;   1       END;



                                                          .PSECT  $PLIT$,NOWRT,NOEXE,2

           4D  45  54  53  59  53  00148 P.ABH:  .ASCII  \SYSTEM\                                          ;
                                   0014E         .BLKB   2
                         00000006  00150 P.ABG:  .LONG   6                                                 ;
                         00000000' 00154         .ADDRESS P.ABH                                            ;
               52  45  4E  57  4F  00158 P.ABJ:  .ASCII  \OWNER\                                           ;
                                   0015D         .BLKB   3
                         00000005  00160 P.ABI:  .LONG   5                                                 ;
                         00000000' 00164         .ADDRESS P.ABJ                                            ;
               50  55  4F  52  47  00168 P.ABL:  .ASCII  \GROUP\                                           ;
                                   0016D         .BLKB   3
                         00000005  00170 P.ABK:  .LONG   5                                                 ;
                         00000000' 00174         .ADDRESS P.ABL                                            ;
               44  4C  52  4F  57  00178 P.ABN:  .ASCII  \WORLD\                                           ;
                                   0017D         .BLKB   3
                         00000005  00180 P.ABM:  .LONG   5                                                 ;
                         00000000' 00184         .ADDRESS P.ABN                                            ;

                                         SYSTEM_DESC=       P.ABG
                                         OWNER_DESC=        P.ABI
                                         GROUP_DESC=        P.ABK
                                         WORLD_DESC=        P.ABM
```

```
                                                         .PSECT  $CODE$,NOWRT,2

                                         07FC 00000 PROTECTION_PARSE:
                                                         .WORD   Save R2,R3,R4,R5,R6,R7,R8,R9,R10                  ; 1236
                          5A    0000V CF  9E 00002       MOVAB   PARSE_PROTECTION_VALUE, R10                       ;
                          59    0000G CF  9E 00007       MOVAB   CLI$GET_VALUE, R9                                 ;
                          58    0000G CF  9E 0000C       MOVAB   CLI$PRESENT, R8                                   ;
                          57    0000' CF  9E 00011       MOVAB   SYSTEM_DESC, R7                                   ;
                          56    0000' CF  9E 00016       MOVAB   CURR_PROTECTION_OR, R6                            ;
                          5E          08  C2 0001B       SUBL2   #8, SP                                            ;
     08        00         6E          00  2C 0001E       MOVC5   #0, (SP), #0, #8, CLI_DESC                        ; 1288
                                      6E     00023
                      03  AE          02  90 00024       MOVB    #2, CLI_DESC+3                                    ; 1289
                                      57  DD 00028       PUSHL   R7                                                ; 1294
                          68          01  FB 0002A       CALLS   #1, CLI$PRESENT                                   ;
                          23          50  E9 0002D       BLBC    R0, 1$                                            ;
                                      52  D4 00030       CLRL    KEY_DISP                                          ; 1301
     66        04         52 FFFFFFFF 8F  F0 00032       INSV    #-1, KEY_DISP, #4, CURR_PROTECTION_OR             ; 1302
04   A6        04         52          00  F0 0003B       INSV    #0, KEY_DISP, #4, CURR_PROTECTION_AND             ; 1303
                               4080   8F  BB 00041       PUSHR   #^M<R7,SP>                                        ; 1307
                          69          02  FB 00045       CALLS   #2, CLI$GET_VALUE                                 ;
                          08          50  E9 00048       BLBC    R0, 1$                                            ;
                                      52  DD 0004B       PUSHL   KEY_DISP                                          ; 1309
                                 04   AE  9F 0004D       PUSHAB  CLI_DESC                                          ;
                          6A          02  FB 00050       CALLS   #2, PARSE_PROTECTION_VALUE                        ;
                                 10   A7  9F 00053 1$:   PUSHAB  OWNER_DESC                                        ; 1316
                          68          01  FB 00056       CALLS   #1, CLI$PRESENT                                   ;
                          25          50  E9 00059       BLBC    R0, 2$                                            ;
                          52          04  D0 0005C       MOVL    #4, KEY_DISP                                      ; 1323
     66        04         52 FFFFFFFF 8F  F0 0005F       INSV    #-1, KEY_DISP, #4, CURR_PROTECTION_OR             ; 1324
04   A6        04         52          00  F0 00068       INSV    #0, KEY_DISP, #4, CURR_PROTECTION_AND             ; 1325
                                      5E  DD 0006E       PUSHL   SP                                                ; 1329
                                 10   A7  9F 00070       PUSHAB  OWNER_DESC                                        ;
                          69          02  FB 00073       CALLS   #2, CLI$GET_VALUE                                 ;
                          08          50  E9 00076       BLBC    R0, 2$                                            ;
                                      52  DD 00079       PUSHL   KEY_DISP                                          ; 1331
                                 04   AE  9F 0007B       PUSHAB  CLI_DESC                                          ;
                          6A          02  FB 0007E       CALLS   #2, PARSE_PROTECTION_VALUE                        ;
                                 20   A7  9F 00081 2$:   PUSHAB  GROUP_DESC                                        ; 1338
                          68          01  FB 00084       CALLS   #1, CLI$PRESENT                                   ;
                          25          50  E9 00087       BLBC    R0, 3$                                            ;
                          52          08  D0 0008A       MOVL    #8, KEY_DISP                                      ; 1345
     66        04         52 FFFFFFFF 8F  F0 0008D       INSV    #-1, KEY_DISP, #4, CURR_PROTECTION_OR             ; 1346
04   A6        04         52          00  F0 00096       INSV    #0, KEY_DISP, #4, CURR_PROTECTION_AND             ; 1347
                                      5E  DD 0009C       PUSHL   SP                                                ; 1351
                                 20   A7  9F 0009E       PUSHAB  GROUP_DESC                                        ;
                          69          02  FB 000A1       CALLS   #2, CLI$GET_VALUE                                 ;
                          08          50  E9 000A4       BLBC    R0, 3$                                            ;
                                      52  DD 000A7       PUSHL   KEY_DISP                                          ; 1353
                                 04   AE  9F 000A9       PUSHAB  CLI_DESC                                          ;
                          6A          02  FB 000AC       CALLS   #2, PARSE_PROTECTION_VALUE                        ;
                                 30   A7  9F 000AF 3$:   PUSHAB  WORLD_DESC                                        ; 1360
                          68          01  FB 000B2       CALLS   #1, CLI$PRESENT                                   ;
                          25          50  E9 000B5       BLBC    R0, 4$                                            ;
                          52          0C  D0 000B8       MOVL    #12, KEY_DISP                                     ; 1367
```

```
        66              04              52 FFFFFFFF    8F   F0 000BB          INSV    #-1, KEY DISP, #4, CURR PROTECTION OR          ; 1368
     04 A6              04              52             00   F0 000C4          INSV    #0, KEY_DISP, #4, CURR_PROTECTION_AND          ; 1369
                                                       5E   DD 000CA          PUSHL   SP                                             ; 1373
                                                 30    A7   9F 000CC          PUSHAB  WORLD DESC                                     ;
                                        69             02   FB 000CF          CALLS   #2, CLI$GET_VALUE                              ;
                                        08             50   E9 000D2          BLBC    R0, 4$                                         ;
                                                       52   DD 000D5          PUSHL   KEY_DISP                                       ; 1375
                                                 04    AE   9F 000D7          PUSHAB  CLI_DESC                                       ;
                                        6A             02   FB 000DA          CALLS   #2, PARSE_PROTECTION_VALUE                     ;
                                                            04 000DD 4$:      RET                                                    ; 1380

; Routine Size:  222 bytes,    Routine Base:  $CODE$ + 0888
```

```
1 ROUTINE PARSE_PROTECTION_VALUE( DESC : REF $BBLOCK,
1                                   FIELD_LOCATION ) : NOVALUE =
1
1 !++
1 !
1 ! FUNCTIONAL DESCRIPTION:
1 !
1 !       This routine parses the keyword value given by the /PROTECTION
1 !       qualifier.  (/PROTECTION=(s:rewd))
1 !
1 ! FORMAL PARAMETERS:
1 !
1 !       DESC            the address of a descriptor which points to the
1 !                       keyword value.
1 !       FIELD_LOCATION  the offset of the appropriate protection field
1 !
1 ! IMPLICIT INPUTS:
1 !
1 !       None
1 !
1 ! IMPLICIT OUTPUTS:
1 !
1 !       bits will be set in CURR_PROTECTION_OR
1 !
1 ! ROUTINE VALUE:
1 !
1 !       None
1 !
1 ! COMPLETION CODES:
1 !
1 !       None
1 !
1 ! SIDE EFFECTS:
1 !
1 !       None
1 !
1 !--
1
2 BEGIN
2
2 LOCAL
2     RTN_STATUS,                       ! status returned from external calls
2     BIT_DISP,                         ! Location of bit to be set in protection  field
2     CHAR_DESC : $BBLOCK[ DSC$C_S_BLN]  ! A descriptor
2     ;
2
2
2 ! The descriptor points to only one character at a time.
2 !
2 CH$FILL( 0, DSC$C_S_BLN, char_desc);
2 CHAR_DESC[ DSC$W_LENGTH ] = 1;
2
2
2 ! Process the keyword value one character at a time.
2 !
2 INCR INDEX FROM 0 TO .DESC[ DSC$W_LENGTH ]-1 DO
3     BEGIN
```

```
;  910          1438  3
;  911          1439  3        CHAR_DESC[ DSC$A_POINTER ] = .DESC[ DSC$A_POINTER ] + .INDEX;
;  912          1440  3
;  913          1441  3        ! Look up the keyword in the keyword table.
;  914          1442  3        !
;  915          1443  4        IF NOT (RTN_STATUS = LIB$LOOKUP_KEY( CHAR_DESC, COPY$PROT_VALUE, BIT_DISP) )
;  916          1444  3        THEN
;  917          1445  4            BEGIN
;  918          1446  4
;  919          1447  4            ! No character match was found, signal the error and return to caller.
;  920          1448  4            !
;  921          1449  4            PUT_MESSAGE( MSG$_BADVALUE, 1, .desc );
;  922          1450  4            RETURN;
;  923          1451  3            END;
;  924          1452  3
;  925          1453  3        ! Clear the mask bit which corresponds to the protection attribute.
;  926          1454  3        !
;  927          1455  3        CURR_PROTECTION_OR[prot_mask( .FIELD_LOCATION + .BIT_DISP, 1)] = NO;
;  928          1456  3
;  929          1457  2        END;                                              ! End of single character value loop.
;  930          1458  1 END;                                                     ! End of routine PARSE_PROTECTION_VAL



                                                 003C 00000 PARSE_PROTECTION_VALUE:
                                                                   .WORD     Save R2,R3,R4,R5                                  ; 1381
                               5E            0C  C2 00002          SUBL2     #12, SP                                           ;
        08          00         6E            00  2C 00005          MOVC5     #0, (SP), #0, #8, CHAR_DESC                       ; 1430
                                      04     AE     0000A                                                                      ;
                         04    AE            01  B0 0000C          MOVW      #1, CHAR_DESC                                     ; 1431
                               54     04     BC  3C 00010          MOVZWL    @DESC, R4                                         ; 1436
                               52     04     AC  D0 00014          MOVL      DESC, R2                                          ; 1439
                               53            01  CE 00018          MNEGL     #1, INDEX                                         ; 1455
                                             40  11 0001B          BRB       3$                                                ;
                         08    AE     04 B243    9E 0001D 1$:      MOVAB     @4(R2)[INDEX], CHAR_DESC+4                        ; 1439
                                             5E  DD 00023          PUSHL     SP                                                ; 1443
                                   0000G     CF  9F 00025          PUSHAB    COPY$PROT_VALUE                                   ;
                                      0C     AE  9F 00029          PUSHAB    CHAR_DESC                                         ;
                  00000000G    00            03  FB 0002C          CALLS     #3, [IB$LOOKUP_KEY                                ;
                               55            50  D0 00033          MOVL      R0, RTN_STATUS                                    ;
                               19            55  E8 00036          BLBS      RTN_STATUS, 2$                                    ;
                                      04     AC  DD 00039          PUSHL     DESC                                              ; 1449
                                             01  DD 0003C          PUSHL     #1                                                ;
                               7E   1114     8F  3C 0003E          MOVZWL    #4372, -(SP)                                      ;
                      0000G    CF            01  FB 00043          CALLS     #1, COPY$MSG_NUMBER                               ;
                                             50  DD 00048          PUSHL     R0                                                ;
                  00000000G    00            03  FB 0004A          CALLS     #3, LIB$STOP                                      ;
                                                 04 00051          RET                                                         ; 1445
                 50      08    AC            6E  C1 00052 2$:      ADDL3     BIT_DISP, FIELD_LOCATION, R0                      ; 1455
                 00   0000'    CF            50  E5 00057          BBCC      R0, CURR_PROTECTION_OR, 3$                        ;
                 BC            53            54  F2 0005D 3$:      AOBLSS    R4, INDEX, 1$                                     ; 1436
                                                 04 00061          RET                                                         ; 1458

; Routine Size:  98 bytes,    Routine Base:  $CODE$ + 0966
;  931          1459  1
;  932          1460  1 END
;  933          1461  0 ELUDOM



                                                                          .EXTRN  LIB$SIGNAL, LIB$STOP

;                                          PSECT SUMMARY
;
;       Name                          Bytes                           Attributes
;
;  $PLIT$                                  392  NOVEC,NOWRT,  RD ,NOEXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
;  $GLOBAL$                                 28  NOVEC,  WRT,  RD ,NOEXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
;  $CODE$                                 2504  NOVEC,NOWRT,  RD ,  EXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)




;                                    Library Statistics
;
;                                                 -------- Symbols --------        Pages       Processing
;       File                                      Total    Loaded   Percent       Mapped       Time
;
;  _$255$DUA28:[SYSLIB]STARLET.L32;1               9776        65         0          581         00:01.0





;                                           COMMAND QUALIFIERS

;       BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/LIS=LIS$:COPYCLI/OBJ=OBJ$:COPYCLI MSRC$:COPYCLI/UPDATE=(ENH$:COPYCLI)

; Size:          2504 code + 420 data bytes
; Run Time:         00:41.2
; Elapsed Time:     01:25.4
; Lines/CPU Min:     2129
; Lexemes/CPU-Min: 21307
; Memory Used:  369 pages
; Compilation Complete
```

0067 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY